Modeling, Analysis and Control of Nonlinear Switching Systems

A Thesis
Presented to
The Academic Faculty

by

Niket S. Kaisare

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

School of Chemical and Biomolecular Engineering
Georgia Institute of Technology
November 2004

Modeling, Analysis and Control of Nonlinear Switching Systems

Approved by:

Jay H. Lee, Advisor
School of Chemical Engineering
Georgia Institute of Technology

Matthew J. Realff
School of Chemical Engineering
Georgia Institute of Technology

Andrei G. Fedorov
School of Mechanical Engineering
Georgia Institute of Technology

Martha Gallivan
School of Chemical Engineering
Georgia Institute of Technology

Athanasios Nenes
School of Chemical Engineering
Georgia Institute of Technology

Date Approved: 22 November 2004

To my grandmothers

Kashibai Kaisare and late Usha Jayakar

ACKNOWLEDGEMENTS

As I look back at this important juncture of my life, I find myself indebted to a number of people, without whom I could not have possibly come this far. Without a doubt, the first person who comes to mind is my thesis advisor Professor Jay H. Lee, for all the guidance, advice, understanding and support that he has given me in the past five-something years. He has been a great motivation for me in pursuit of academic excellence and has taught many things, well beyond the realms of my thesis work. Appreciation of Korean food is definitely one of them.

I am also thankful to Professor Andrei G. Fedorov for the immense help, encouragement and guidance that he provided me as a committee member. As a matter of fact, he was more like a co-advisor to me than a committee member. I appreciate the numerous times he has gone out of his way to help me. I find his enthusiasm very infective. The help I received from the other committee members Professors Athanasios Nenes, Matthew Realff and Martha Gallivan is also greatly appreciated. I thank all these professors for agreeing to be on my thesis committee and giving me valuable advice, especially during the latter half of my PhD.

I don't think I would have made it through if not for the help and friendship of the various members of the Lee research group. I would especially like to thank Jong Min Lee, with whom I collaborated several times, and Jaein Choi, who tolerated me as a lab mate for the last two years (it is actually debatable if he tolerated me or vice versa). I have always looked upon these two as some sort of a "yardstick" to measure my own performance, and have always derived motivation from their hard work and dedication. Not to forget, the numerous lunch and dinner trips, especially to the Korean restaurants. I would also like to thank Seshatre Natarajan, Andrew Dorsey and Yang Dong Pan for being great seniors to me. And to Tina Tosukhowong, Manish Gupta, Swathy Ramaswamy, Anshul Dubey, and Nikolaos Pratikakis for being nice and responsible junior students to work

with. Especially Seshatre and Manish have been great friends both within and outside the workplace. During my stay, I have had a great opportunity to interact with various visitors, Jochen Till, Professor Dae Ryook Yang, Dr. Jong Ku Lee and several others.

I have been very fortunate to have had excellent roommates throughout my stay in Atlanta. Sushanta Mohapatra is like an elder brother to me. His wife Kavitha did in two months what my mother could not in twenty one years: make me eat spinach. I am so grateful for their love and affection. What can I say about Mudasir Ahmad, the guy who flew all the way from San José, CA to attend my defense? I still remember those journeys to Atlanta diner at 4 or 5 am to have a quick bite before calling it a night. Shabbir has also been a great friend and roommate for the last year.

Beyond these, I have made a great number of friends in Georgia Tech, with whom I shared many a good time. Ashwini and Preeti were always there to help me; I don't know how I could have survived the last six months without them. I have met many nice people and made great friends in Georgia Tech and Purdue. Since this is not an autobiography, I will not mention their names.

Finally, I would like to thank my grandparent, my parents, my aunts Shobhana and Kamal, my uncle Mohan and my cousin Umesh for their love and support. I could not have possibly come this far had it not been for their strength and their belief in my abilities.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

NOMENCLATURE

Part I: Periodic Port Switching

$\underline{a}$	surface area per unit volume of solid	m^2/m_s^3
$\hat{\underline{a}}$	surface area per unit void volume	m^2/m_g^3
c_{pg}	average specific heat of the gas	$J/(kg.K)$
c_s	specific heat of solid	$J/(kg.K)$
C	concentration	mol/m^3
D_{im}	diffusivity	m^2/s
E	activation energy	J/mol
F	radiation view factor	—
h_v	heat transfer coefficient	$W/(m^2.K)$
k_0	Arrhenius rate constant	$(mol/m^3)^{-\iota} sec^{-1}$
k_{gi}	mass transfer coefficient	m/s
l	reactor length	m
M_i	molecular weight	$kmol/kg$
n_{rxn}	# of reactions	—
n_{sp}	# of species	—
p	pressure	bar
q_{rad}	radiation heat flux	$W/(m^2.K)$
r	reaction rate	$mol/(m^2$ surface.$s)$
R	gas constant	$J/(mol.K)$
T	temperature	K
v	gas velocity	m/s
ΔH	heat of reaction	J/mol

Greek Notations

α	coefficient to split reaction heat between gas and solid	—
ϵ	Emissivity of the reactor wall	—
λ	thermal conductivity	$W/(m.K)$
ν	stoichiometric coefficient	—
ϕ	any state variable	units vary
ρ	density	kg/m^3

Subscripts and superscripts

0	condition at reactor inlet
i	i^{th} species
j	j^{th} reaction
g	gas phase
s	solid surface

Part II: Switching Between Multiple Steady States

Controller

c	biomass concentration
c_{SP}	setpoint
D	dilution rate
h	sampling interval
J	cost-to-go function or value
$\tilde{J}$	neural network expressing cost-to-go function
J_i	cost incurred in going from state x_{p-i} to terminal state x_p
J^i	cost-to-go function for i^{th} iteration
J^i_j	cost-to-go function for i^{th} iteration corresponding to the state x_j
p	control horizon
Q	output penalty weight

R	input penalty weight
s_i	concentration of i^{th} substrate
s_{if}	feed concentration of i^{th} substrate
u	manipulated variable
x	state vector
x_p	terminal state
ϕ	single stage cost
$\bar{\phi}$	terminal cost

Cybernetic model

e_i	concentration of i^{th} key enzyme
K_{ei}	saturation constant for expression of i^{th} key enzyme
K_i	saturation constant for i^{th} reaction
r_{ei}	enzyme expression rate for i^{th} reaction
r_{ei}^*	constitutive rate of enzyme expression for i^{th} reaction
r_{ei}^{max}	maximum rate enzyme expression rate for i^{th} reaction
r_i	rate of i^{th} reaction
r_i^{max}	maximum rate of i^{th} reaction
r_g	growth rate
u_i	cybernetic regulation variable governing expression of i^{th} key enzyme (Note: not to be confused with manipulated variable u)
v_i	cybernetic regulation variable governing activity of i^{th} key enzyme
Y_i	yield coefficient for i^{th} reaction
β_i	turnover rate for i^{th} key enzyme

SUMMARY

The first part of this two-part thesis examines the reverse-flow operation of autothermal methane reforming in a microreactor. Most of the conventional processes are run under unidirectional steady state conditions. However, forced unsteady state operation can provide significant advantages over steady state operation. A reverse-flow operation, where the input and output ports are periodically switched resulting in a periodic reversal of flow direction, exploits the dynamic nature of catalytic processes and thermal inertia of the reactor to give better reactor performance. The performance improvement in a reverse-flow reactor may be due to one or more of the following reasons: favorable thermodynamic conditions for water-gas shift reaction at the exit, improved thermal utilization of the reactor, and using the reactor as a regenerative heat exchanger.

In this thesis, a theoretical study is undertaken to explain the physical origins of the experimentally observed improvements in the performance of the reverse-flow operation of a microreactor. First, a scaling analysis is presented to understand the effect of various time scales existing within the reactor, and to obtain guidelines for the optimal reverse-flow operation. Then, the effect of kinetic parameters, transport properties, reactor design and operating conditions on the reactor operation is parametrically studied through numerical simulations. The reverse-flow operation is shown to be more robust than the unidirectional operation for variations in these system parameters, with respect to both optimal operating conditions as well as variations in hydrogen throughput requirements. A rational scheme for improved catalyst placement in the microreactor, which exploits the spatial temperature profiles in the reactor is also presented. Finally, a design modification called the "opposed flow" system, similar to countercurrent heat exchanger with reacting streams flowing in adjacent channels, is suggested. The opposed flow microreactor retains the performance benefits of the reverse-flow operation without requiring the input / output port switching, thereby potentially simplifying practical implementation.

In the second part of this thesis, a novel simulation-based Approximate Dynamic Programming framework is presented for optimal control of steady state switching in a bioreactor. The multiple steady states are characterized by different metabolic states within the bacterial cells. The cybernetic modeling framework is used to capture these cellular metabolic switches. These switches are associated with significant nonlinearities and non-differentiability (discontinuity in the first derivative) of the model equations. Additionally, the bioreactor exhibits long-lived transient dynamics (ie. small variations in the system state for extended periods of time) punctuated by relatively fast-time metabolic switches. Model Predictive Control (MPC), one of the most popular advanced control methods, is able to drive the system to the desired set-point. However, the nonlinearity and non-differentiability cause computational problems with MPC. Furthermore, the MPC solution is suboptimal due to using shorter time horizons in the presence of long-lived dynamics. Approximate Dynamic Programming has an advantage over MPC as the closed-loop optimal policy is computed *offline* in the form of so called "value" or "cost-to-go" function. Value function is a function of the system state and it expresses the long-term desirability of each state. Through the use of the value function, the infinite horizon problem is converted into an equivalent single-stage problem, which can be solved online.

The aim of Approximate Dynamic Programming is to obtain an approximation to the value function using closed-loop simulations under suboptimal policies, function approximation to express cost-to-go values as a function of the system state, and learning algorithms to iteratively improve the cost-to-go approximation until convergence. The offline learning and online control are restricted to the subset of the state space defined by the points visited by suboptimal simulations. The function approximation is used to interpolate cost-to-go values within this visited subset. This method is applied to the steady state switching problem. Extrapolation to the unvisited regions is shown to provide performance problems. Hence different methods to restrict the state trajectories to the visited subset, and to expand the subset through systematic exploration of the state space are considered. A comparison of various function approximators used within this framework is presented. Finally, two popular algorithms for obtaining an optimal value or cost-to-go function, viz. value iteration

and policy iteration are compared.

Following peer reviewed publications resulted from this work:

- Kaisare N. S., J. H. Lee and A. G. Fedorov (2004), "Hydrogen generation in a reverse-flow microreactor: 1. Model formulation and Scaling," *A.I.Ch.E. Journal*, accepted.

- Kaisare N. S., J. H. Lee and A. G. Fedorov (2004), "Hydrogen generation in a reverse-flow microreactor: 2. Simulation and analysis," *A.I.Ch.E. Journal*, accepted.

- Kaisare N. S., J. H. Lee and A. G. Fedorov, "Operability and sensitivity analysis of a reverse-flow microreactor," *in preparation*

- Kaisare N. S., J. M. Lee and J. H. Lee (2003), "Simulation based strategy for nonlinear optimal control: Application to a microbial cell reactor," *International Journal of Robust and Nonlinear Control*, 13: 347-363

- Lee J. M., N. S. Kaisare and J. H. Lee (2004), "An evolutionary improvement of control policies via approximate Dynamic Programming: Part 1: A comparative study on the choice of approximator," *Journal of Process Control*, submitted

- Kaisare N. S., J. M. Lee and J. H. Lee, "Empirical results on convergence and exploration in simulation-based Approximate Dynamic Programming," *IFAC World Congress* 2005, submitted[1]

[1] An extended version of this proceedings will be prepared for submission to a peer-reviewed journal.

CHAPTER I

INTRODUCTION

1.1 Background

There has been a lot of interest in the study of hybrid dynamical systems in recent years, motivated by a growing number of practical systems involving interaction between continuous variables and discrete events / decisions. Hybrid dynamical systems are the systems that involve both continuous and discrete dynamics. Branicky et al. [19] presented a unified framework for analysis and control of hybrid systems. Switching systems can be considered as a subclass of hybrid systems. By switching system, we refer to the case where the underlying model is a continuous model, with an "event" triggering a "switch." In general, a hybrid system may be represented as

$$x(t+1) = f_i(x(t), u(t), w(t); \theta_i(t)) \tag{1}$$

Here, each vector field $f_i(.)$ represents a continuous-time model of the system. A switching is triggered at discrete intervals, represented through binary variables θ_i.

The switching systems can be classified based on the "trigger" as:

- **External or input-driven switching**: This case arises when external inputs to the system undergo a switch. In a number of cases, the time of switching is either known or is a decision variable. An example of this is the reverse-flow operation, where the input / output ports are periodically switched. Another example is the gear shifting in an automobile. Disturbance switching can also be classified in this category. For example, the change from soft-wood to hard-wood chips in a pulp digester introduces abrupt jumps in disturbance characteristics.

- **Internal or state-driven switching** This is also known as *autonomous switching*, wherein the switching is triggered when state vector hits certain boundaries. Linear

hybrid systems, piece-wise affine systems and switching between multiple metabolic states of a bioreactor are examples of autonomous switching. Another example of autonomous switching is collision between two bodies.

The switching systems can also be classified based on the type of response of the system to the trigger as follows:

- **Jump**: In this case, the state or input vector itself switches, but the same vector field $f_i(.)$ is applicable. An example of this is collision between two bodies, wherein the velocity instantaneously switches but the underlying dynamical model remains the same.

- **Impulse**: In case of an impulse, the underlying system model $f_i(.)$ itself switches, such as that seen in piecewise affine models.

In this thesis, we do not aim to obtain a unified method for analysis or control of switching systems. Instead, two examples of switching systems are considered separately in the two parts of the thesis. These examples are motivated by experimental work on a reverse-flow reactor [59] and a continuous bioreactor with multiple steady states [32]. Part-I of this thesis considers modeling, analysis and design of the reverse-flow microreactor. The flow direction is periodically reversed by switching the input / output valves, which is modeled as *input-driven jumps* in the system. Part-II of this thesis considers optimal control of switching between multiple metabolic states in a bioreactor. The steady state switching is accompanied by changes in the metabolic state of the biomass. In different metabolic states, different metabolic pathways are fully activated or deactivated, represented in the model by the so-called cybernetic regulation variables that switch between the values of $v_i = 1$ for fully active and $v_i = r_i/r_{max}$ for deactivated states. Thus, metabolic switches are examples of state-dependent or *autonomous impulses*.

1.2 Motivation and Past Work

1.2.1 Reverse-Flow Microreactor

Last few years have seen significant advances in Micro Electro Mechanical Systems (MEMS) technology including applications to chemical reaction engineering. Following advances in μTAS (micro Total Analysis Systems), it is now possible to produce microsystems with complex geometries, flow patterns, and spatial placement of catalyst, sensor and actuator elements [110]. The microreactor technology has several advantages for chemical production [29, 54]. These include increased heat and mass transfer rates, possibility of performing reactions under more aggressive conditions, allowing new reaction pathways and integration of sensors and actuators. A number of examples of microreactors were developed to use for generation of hydrogen-for-fuel-cells using partial oxidation [95], ammonia decomposition [40], water gas shift reactions [85], etc. [54]. However, most of these papers have looked at steady state operation of the reactors. Only recently have forced unsteady state applications of the microreactors, such as concentration cycling [92], temperature cycling [18] or reverse-flow operation [59], been reported.

On the other hand, forced unsteady state operation of well-mixed [6] and fixed bed systems [50] has been an area of great interest over three decades [72, 73, 30, 74, 94]. The significant know-how and theoretical studies on this subject is largely attributed to the Novosibirsk group of Matros [73]. Recently, Silveston and Hudgins [97] reviewed state of the art on periodic temperature cycling in a reactor. They reported that although a large body of theoretical results exist, practical applications of fast temperature switching in conventional reactors have been elusive. Due to the high inertia of the conventional reactors, the attainable range of frequencies is rather low and in the order of $10^{-4} - 10^{-2}$ Hz [92]. As a result, theoretically predicted performance improvements could not be obtained in the conventional reactors. However, lower thermal inertia, faster heat (and mass) transfer rates and significantly faster dynamics of a microreactor means that fast switching of the reactor is possible, and thus theoretically predicted improvements may become possible. Rouge et al. [92] were able to obtain theoretically predicted improvement in the rates of dehydration of isopropanol using concentration cycling, while Brandner et al. [18] were able to provide

temperature modulation employing 5 seconds cycles at $50°C$ amplitude, something that is not possible in a macro-reactor.

Since a large body of theoretical and experimental investigation of reverse-flow operation exists for fixed bed reactor, it remains to be understood how these results translate to a microreactor. The salient features of a microreactor are high rates of heat and mass transfer, lower thermal inertia and faster response times. High frequency switching is possible in a microreactor to reap benefits of unsteady state or reverse-flow operations. Additionally, different reactor configurations that yield performance equivalent to infinitely fast switching can be produced and tested. Recently, Sheintuch and Nekhamkina [96] compared the reverse-flow operation with an internal loop reactor and a countercurrent reactor on a conventional (macro-) scale. Elaborate reactor / heat exchanger designs were required to ensure sufficient heat transfer between the cold inlet and the hot effluent streams. In contrast, heat transfer rates in a microreactor are very high. Thus, theoretical and experimental investigation into the reverse-flow operation is required to corroborate or modify the unsteady state operation results for larger scale reactors.

This work draws its motivation from an experimental study, where Kikas et al. [59] showed that hydrogen productivity can be increased by operating the microreactor in a reverse-flow manner. Their results and related work from the literature is further reviewed in individual chapters.

1.2.2 Control of steady state switching in a bioreactor

Steady state multiplicity is the condition in which a reactor exhibits two or more output conditions for the same set of input conditions. A prerequisite for steady state multiplicity is the nonlinearity of a system. A simplest example is that of an adiabatic reactor carrying out exothermic reactions, wherein the multiplicity arises due to the nonlinear dependence of reaction rate on temperature [37]. Typically, in systems with multiple steady states, one of the states is the desired state. Often, control of the system at the steady state becomes important because the alternate steady states can either result in reactor quenching, or cause a runaway conditions in the worst case, or result in a significant loss of productivity

towards the desired product.

Recently, Hu and coworkers [32], and Stephanopolous and coworkers [36] reported striking cases of steady state multiplicity in continuous cultures of mammalian hybridoma cells. Namjoshi et al [79] modeled the system of Europa et al. [32] and presented a nonlinear bifurcation analysis of the same [80]. Experimental and model results show that the desired steady state can be obtained only by carefully starving the biomass so that they direct their metabolism towards producing the desired product instead of the waste intermediates. Namjoshi et al. [79] also used their model to numerically determine improved feeding profiles for the start up of the reactor in the desired steady state. The feed rates varied significantly with variations in the inlet concentrations of the reactor. Due to the propensity of the hybridoma cells to prefer the (undesirable) low yield steady state, a control scheme that drives the reactor to the desired steady state is needed.

While the work on hybridoma cultures show a striking case, other examples of have been found [81, 82]. One of the simplest bioreactor to display steady state multiplicity is a chemostat containing bacterium growing simultaneously on two substitutable substrates [62]. While control of continuous bioreactors using cybernetic model predictive control (MPC) has been considered [39], control of multiple steady states in bioreactors has not been studied yet.

The control of bioreactors is made difficult by the nonlinear dynamics, operation constraints, lack of reliable models and long-ranged transients [22]. As a result, model-based control schemes such as MPC suffer from significant computational burden in solving the control problem online. Hence, linear assumptions are employed or shorter horizons are considered, leading to a loss in the control performance. Only recently have our colleagues started developing simulation-based approximate Dynamic Programming methods as an alternative to MPC [66]. This method reduces the infinite horizon problem in MPC to an equivalent one-stage problem through the use of the so-called "*cost-to-go* approximator." The cost-to-go is calculated offline for all the relevant states in the state space, and is used online to reduce the computational burden and improve the optimality of optimal control. Specific discussions on this method and related work from the literature will be deferred

until Chapter 6.

1.3 Thesis Objectives and Outline

This objectives of this thesis are two-fold: (1) to provide a mathematical framework for analyzing operation of a microreactor with periodic switching of flow direction, and (2) to develop an algorithmic framework for control of switching between multiple steady states in a bioreactor. The specific goals of this thesis are posed in the form o f following questions:

- What are the physical / chemical origins of the performance improvement of a reverse-flow microreactors over unidirectional ones?
- Can we define the guidelines for optimal operation of a reverse-flow reactor using simple scaling rules, by relating the reactor performance with time scales of individual processes taking place within the reactor?
- What is the importance of transport properties, reaction parameters, and thermo-physical properties of the reactor structural support? How does the optimal reactor operation depend on these parameters?
- How does one obtain robust operation by operating in the reverse-flow mode, as compared to the unidirectional mode, and what are the performance limits? Can we make design changes to make the unidirectional operation more robust?
- How do the methods used in Reinforcement Learning (RL) and Neuro-Dynamic Programming (NDP) literature to solve the DP problem in an approximate sense, translate for optimal chemical and biochemical control problems?
- What are the issues involved in the simulation-based Approximate DP framework? What are the options for improving the function approximator in ADP; how do the various learning algorithms compare; what restrictions are brought about by working in the visited subset of the state space and how can we systematically expand this subset?

The rest of the thesis is organized as follows. The case study of hydrogen generation by methane partial oxidation in a microreactor with periodic port switching is presented in Part-I. In chapter 2, a one-dimensional (1-D) model of the microreactor is developed; a critical comparison of methane oxidation and reforming kinetics from the literature is performed; an analysis of the time scales of individual processes within the reactor is presented in order to gain fundamental insight into the reactor operation; finally, the effect of radiation heat transfer is also considered. Chapter 3 presents parametric simulation and analysis for the reverse-flow operation of the microreactor. The effect of varying operating conditions such as the inlet $CH_4 : O_2$ feed ratio, the inlet velocity and the inlet gas temperature, as well as design parameters such as the reactor length and heat losses to the ambient are parametrically investigated. The issue of choosing the optimal switching time is also addressed in order to obtain design and operation guidelines based on simple time-scale analysis. Chapter 4 presents sensitivity and operability analysis of the reactor. The sensitivity to variations in the transport parameters and the kinetic rate constants is presented and a method for improved placement of various catalysts in the reverse-flow reactor is considered. Next, the reactor operability and robustness is analyzed and operating diagrams for autothermal unidirectional and autothermal reverse-flow operation are presented; and design changes to improve the robustness of unidirectional operation are suggested.

Part-II of this thesis considers control of optimal switching between two different steady states within a continuous bioreactor using the Approximate Dynamic Programming method. Chapter 5 introduces the problem of switching between multiple steady states in continuous microbial and mammalian reactors, and presents successive linerization-based Model Predictive Control algorithm for control of steady state switching. Motivated by the need to reduce computational load and improve the performance of MPC, the approximate DP framework is employed in Chapter 6. A comparison between the online performance of the two methods (ADP and MPC) is presented. Finally, Chapter 7 addresses several specific issues in implementation of ADP, viz. the choice of function approximator, comparison of the learning algorithms and the issue of judicious exploration in the state space.

Finally, the contributions of this thesis and suggestions for the future work are summarized in Chapter 8.

PART I

Periodic Port Switching

CHAPTER II

HYDROGEN GENERATION IN A REVERSE-FLOW MICROREACTOR: MODEL FORMULATION AND SCALING

2.1 Introduction

Conversion of methane or natural gas to synthesis gas via endothermic steam reforming or autothermal partial oxidation has been investigated extensively [7]. The changing geopolitical situation and the growing need to power emerging civilian, space or military applications have ensured that hydrogen generation remains an active research area, although the technology, scales of operation and the end use keeps changing. Traditionally, hydrogen is considered a useful raw material for manufacture of ammonia, urea, methanol and other petrochemical applications. Following the energy crisis in the 1970s and more recently the Gulf wars, conversion of synthesis gas to liquid fuels using Fischer-Tropsch synthesis gained significant attention [1, 103]. More recently, hydrogen is being used in fuel cells for applications varying from powering portable electronic devices, to providing vehicle motive power to large scale power generation [48].

Two main types of micro fuel cells currently researched as possible replacements for the existing portable energy sources are hydrogen-based fuel cells and direct methanol-based fuel cells. One of the important challenges in making the former a viable technology is the on-demand and energy efficient generation of hydrogen. Successful commercialization of this technology requires the reactor to run unattended for long periods of time and operate *autothermally*, i.e. without external addition of heat. Recently, Fedorov and coworkers [59] demonstrated experimentally a possibility of autothermal reforming of methane in a tubular microreactor, when run in either unidirectional (UD) or reverse-flow (RF) mode. They showed that running the reactor with periodic reversal of flow direction results in an

increase in the hydrogen yield compared to the unidirectional operation. The reactor feed consisted of methane and oxygen at room temperature and Pt wire was used as a catalyst.

In this chapter, we develop a one-dimensional (1-D) model that combines heat and mass transfer with heterogeneous catalytic reactions for the reactor of Kikas et al. [59]. We first present the formulation of the model equations and selection of reaction kinetics for partial oxidation. Currently, there is no consensus whether the reactions proceed via direct partial oxidation [47] or via indirect oxidation-reforming steps [43]. In this paper, the latter reaction chemistry comprised of oxidation, reforming and water-gas shift reactions is assumed and the kinetic equations developed by Gosiewski et al. [43] are used. The sensitivity to the kinetic equations is demonstrated by comparing the above kinetics with those considered by de Smet et al. [26]. The main reaction generating hydrogen is the endothermic reforming reaction ($CH_4 + H_2O \rightleftharpoons CO + 3H_2 \quad \Delta H^0_{298} = 206.2 kJ/mol$), which is catalyzed at elevated temperatures, and therefore requires addition of heat to sustain the reaction. In autothermal operation, this heat is provided by the highly exothermic oxidation of methane ($CH_4 + 2O_2 \rightleftharpoons CO_2 + 2H_2O \quad \Delta H^0_{298} = -802.3 kJ/mol$). The overall global reaction occurring in the reactor ($CH_4 + 0.5O_2 \rightleftharpoons CO + 2H_2 \quad \Delta H^0_{298} = -35.7 kJ/mol$) being only slightly exothermic, reverse flow reactors have been used to maintain autothermal operation of the reactors by using the catalyst bed as a regenerative heat exchanger [14, 25].

The comparison of UD and RF operation of the reactor is the main focus of this study. Blanks et al. [14] provided one of the first experimental and simulation study of autothermal methane partial oxidation in a pilot-scale fixed bed reverse-flow reactor (RFR). Later, de Groote et al. [25] and Gosiewski et al. [43] performed simulation study of methane partial oxidation in a fixed bed RFR. In both cases, a thermal wave with moving reaction zone was observed in UD operation; thus a RF operation was required to trap the thermal wave within the reactor and maintain autothermal operation. This study differs from the above mentioned papers on two accounts. First, the reactor being a microchannel reactor, reagents need to diffuse over a shorter distance. The reactions may therefore proceed close to their intrinsic rates. The catalyst loading in this reactor is significantly lower than a fixed bed reactor as the catalyst is not in a well dispersed state. Secondly, autothermal operation was

possible in the microreactor for both UD and RF operation. To our knowledge, this is the first study on the comparison of UD and RF operation of methane partial oxidation in a microreactor. Further, a review of the literature indicates that radiation heat transfer has largely been neglected in previously reported simulations of partial oxidation. Therefore, section 2.5 is devoted to modelling the effect of radiation heat transfer. In spite of high emissive power in the reactor, the overall effect of radiation is demonstrated to be marginal in the longer reactor with high aspect ratio ($l/d = 232$); however, the effect of radiation on a shorter reactor ($l/d = 100$) is quite significant.

2.2 Reactor Model

2.2.1 Description and model equations

In the experiments reported by Kikas et al. [59], the reactor consists of a ceramic tube with four cylindrical channels of 500 microns inner diameter and 11.6 cm length. The center of each reactor channel contains Pt/13%-Rh wire that acts as a catalyst. The ceramic tube is housed in a steel casing with heating elements wrapped around it, which are used to ignite the partial oxidation reactions during reactor startup. The entire setup is insulated to minimize the heat losses. We attempt to analyze numerically these experiments to develop an insight into the advantage offered by RF operation observed in experiments [59].

To this end, a one-dimensional (1-D), two phase model is developed for simulating the autothermal operation of the reactor. The process properties (temperatures, concentrations, etc.) are assumed to be uniform in cross-sectional directions. The flow is one dimensional and laminar, and the reactants and products are assumed to be ideal gases. Heterogeneous reaction chemistry consisting of oxidation, reforming and water gas shift reactions is considered. The pseudo-steady state assumption is employed for species balance on solid surface. Radiation effects are neglected at first, and included in the analysis in the last section of this paper.

The resulting conservation equations for species and energy are:

$$\frac{\partial}{\partial t}C_i + \frac{\partial}{\partial z}vC_i = -k_{gi}\hat{\underline{a}}\,[C_i - C_{is}] \tag{2}$$

$$0 = k_{gi}\left[C_i - C_{is}\right] + \sum_{j=1}^{n_{rxn}} \nu_{ij} r_j \tag{3}$$

$$\frac{\partial}{\partial t}T_g + v\frac{\partial}{\partial z}T_g = \frac{h_v \hat{\underline{a}}}{\rho_g c_{pg}}\left[T_s - T_g\right] + \frac{(1-\alpha)\hat{\underline{a}}}{\rho_g c_{pg}} \sum_{j=1}^{n_{rxn}} (-\Delta H_j) r_j \tag{4}$$

$$\frac{\partial}{\partial t}T_s = \frac{\lambda_s}{\rho_s c_s}\frac{\partial^2}{\partial z^2}T_s - \frac{h_v \underline{a}}{\rho_s c_s}\left[T_s - T_g\right] + \frac{\underline{a}}{\rho_s c_s}\left\{\alpha \sum_{j=1}^{n_{rxn}} (-\Delta H_j) r_j + \dot{q}_{rad} + \dot{q}_{\infty}\right\} \tag{5}$$

In the full 3-D model, the reaction heat term appears as a boundary condition. For 1-D models, its customary to attribute the heat of reaction to the solid heat balance (*i.e.* $\alpha = 1$). However, for the sake of generality, we consider splitting the heat of reaction term between the solid and gas phases in Eq. (4) and (5) through the parameter α.

The mass and heat transfer coefficients are computed using the Nusselt and Sherwood number correlations for laminar flow in cylindrical channels, based on constant heat flux and surface concentration boundary conditions, respectively.

$$N_{Nu} = \frac{h_v d}{\lambda_g} = 4.36 \qquad N_{Sh} = \frac{k_{gi} d}{\mathcal{D}_i} = 3.66$$

The pertinent boundary conditions are as follows:

$$\begin{aligned}
C_i(0,t) &= C_{i0} \\
T_g(0,t) &= T_{g0} \\
\lambda \left.\frac{\partial T_s}{\partial z}\right|_{0,t} &= h_{in}\left[T_s(0,t) - T_{g0}\right] + \epsilon\sigma\left[T_s^4(0,t) - T_{g0}^4\right] \\
\left.\frac{\partial T_s}{\partial z}\right|_{l,t} &= 0
\end{aligned}$$

The mole fractions of methane and oxygen are obtained from the feed ratio (0.7 to 1.2) used in the experiments [59]. All other species are assumed to be present in very small quantities to prevent singularities in numerical computations, as some of them appear as denominators in reaction rates. As initial condition, the reactor is assumed to be preheated to $T_{init} = 1173K$. The nominal operating conditions are summarized in Table 1.

During the reverse-flow operation, the input and output ports of the reactor are switched periodically. In computations, the same governing equations and boundary conditions apply in forward and reverse directions. The states are switched according to

$$\phi(z, m\tau^+) = \phi(l - z, m\tau^-) \tag{6}$$

Table 1: Nominal operating conditions for simulations

Reactor length	l	$11.6\,cm$
Channel diameter	d	$500\,\mu$
Ceramic tube o.d.	d_t	$0.254\,cm$
Number of channels		4
Inlet temperature	T_{g0}	$300\,K$
Inlet velocity	v_0	$1.68\,m/s$
Feed compositions		
	$y_{\mathrm{CH_4}}$	0.5
	$y_{\mathrm{O_2}}$	0.5
	y_i	10^{-8} to 10^{-5}
Preheat temperature	T_{init}	$1173\,K$
Switching time	$\tau_{c/2}$	$5\,sec$

where $\phi(z,t)$ represents any state variable, τ is the half-cycle period and the superscripts $+$ and $-$ represent the system after and before switching.

2.2.2 Computing velocity and pressure fields

In order to solve the model equations (2–5), the velocity field needs to be computed, which is typically done by solving the momentum conservation equation simultaneously with the mass and energy conservation. The 1-D momentum conservation equation is given by

$$\frac{\partial}{\partial t}\rho v + \frac{\partial}{\partial z}\rho v v = -\frac{\partial p}{\partial z} + \mu\frac{\partial^2 v}{\partial z^2} \tag{7}$$

where the ideal gas assumption yields us

$$p = \sum_{i=1}^{n_{sp}} C_i R T_g \tag{8}$$

Solving equations (2–5) and (7) simultaneously is computationally very demanding. As a result, one often resorts to making certain assumptions regarding the velocity field.

One possible assumption is to consider the velocity to be constant along the length of the reactor [90]. In reality, for a compressible fluid under quasi-steady state conditions, the total mass flux remains constant within the reactor. The increasing temperature and species generation by reaction affects the density of the system, which consequently changes the velocity through the continuity equation. As a result, constant velocity assumption may lead to incorrect results.

Other possibility is to use an empirical pressure drop relationship to specify the pressure field within the reactor and then obtain the velocity field using the overall continuity equation [25]. For laminar flow of gases in cylindrical channels, the pressure drop along the reactor length is less than 1% of the total pressure. Indeed, solving the energy, mass and momentum conservation (2–5) and (7) simultaneously revealed that the pressure drop along the length of the reactor is less than 0.1 to 0.5% of the total pressure. Hence, it is reasonable to make an assumption that the pressure is constant along the reactor. Summing the species balance (2) over all species and neglecting the time-dependent term due to very fast mass equilibrium (see time scale analysis in the next section), we obtain the relationship for the change in molar flow rate due to reaction [25] as

$$\frac{\partial F}{\partial z} = -\sum_{i=1}^{n_{sp}} k_{gi}\hat{\underline{a}}[C_i - C_{is}] \tag{9}$$

Coupling (9) with (2–5) and noting that the molar flow rate $F\ mol/(m^2.s)$ is given by

$$F = v\frac{p}{RT_g} \tag{10}$$

the problem formulation becomes complete.

We found that indeed solving (2–5) with (9) gives accurate results, while requiring much lesser computational time. Hence, in the following discussion, we employ the constant pressure assumption and use the continuity equation (9) to obtain the velocity within the microreactor.

2.2.3 Reaction Kinetics

The indirect partial oxidation of methane is considered to take place through the following three reactions [43, 26] on the catalyst surface:

$$\text{Oxidation:} \quad CH_4 + 2O_2 \rightarrow CO_2 + 2H_2O \qquad \Delta H^0 = -802.0\ KJ/mol \tag{11}$$

$$\text{Reforming:} \quad CH_4 + H_2O \rightleftharpoons CO + 3H_2 \qquad \Delta H^0 = +206.1\ KJ/mol \tag{12}$$

$$\text{W. G. Shift:} \quad CO + H_2O \rightleftharpoons CO_2 + H_2 \qquad \Delta H^0 = -41.15\ KJ/mol \tag{13}$$

The other reactions that possibly occur in the system, such as CO_2 reforming, CO oxidation, methane reforming to CO_2 etc. are neglected.

The first set of kinetic equations we consider were developed by Gosiewski et al. [43] for methane partial oxidation and were expressed per kg of catalyst for a dispersed Pt on porous alumina pellets. The catalyst loading in the microreactor used in [59] was computed to be 28.9 mg Pt per channel. However, the catalyst in the microreactor was not dispersed; instead a wire was used. Hence the kinetic equations were modified to account for the fact that only a fraction of Pt catalyst is actually available for surface reactions. We converted the kinetic equations to per m^2 catalyst surface basis. We then noted that the density of catalyst particle for the fixed bed system was $2130\,kg/m^3$, while the density of catalyst wire for our microreactor was $19640\,kg/m^3$. Hence, we multiplied all the kinetic rate constants with a factor of 9.22, which is equal to the density ratio. The resulting kinetic equations are summarized in Table 2. Hereafter, we will call these kinetic equations as GOS model.

The effect of the choice of kinetic equations was also considered. Recently, de Smet et al. [26] compared the reforming kinetics of Xu and Froment [119] (denoted by XF) with those of Numaguchi and Kikuchi [83] (denoted by NK) for indirect partial oxidation of methane. They used oxidation kinetics of Trimm and Lam [108] for both cases. In this paper, the GOS model is compared with the NK model. The XF model could not be used for for the following reason. In the XF model, the rate of reforming reaction has the form

$$r = \frac{k}{p_{\mathrm{H_2}}^{2.5}} \frac{p_{\mathrm{CH_4}} p_{\mathrm{H_2O}} - p_{\mathrm{H_2}}^3 p_{\mathrm{CO}} / K_{eq}}{\left[1 + K_{CH4} p_{CH4} + K_{H2O} p_{H2O} + K_{CO} p_{CO} + K_{H2} p_{H2}\right]^2}$$

As the system contains no hydrogen or water in the inlet, this model is not applicable; even with 2% hydrogen and water present in the feed, negative values of $p_{\mathrm{H_2O}}$ were predicted. In the NK model, on the other hand, the rate of reforming reaction has an inverse dependence on $p_{\mathrm{H_2O}}$. As a result, at higher $CH_4 : O_2$ ratios, the reforming rate is non-zero even at zero H_2O concentrations. Hence, a "saturation" term was added to the denominator to ensure that the rate approaches 0 as $p_{\mathrm{H_2O}} \to 0$. Another modification was that instead of oxidation kinetics for Pt on porous alumina, the kinetics published for the non-porous catalyst [108] was used. The procedure suggested in [26] was followed to obtain rate expressions shown in Table 3. In these expressions, the equilibrium constant is computed using the

thermodynamic relationship

$$\Delta G_j = -RT_s \ln(K_{eq,j}) \tag{14}$$

where ΔG_j, the free energy change for the reaction j, is computed using standard thermodynamic correlations obtained from ChemKin database[1]. The value of K_{sat} was chosen to be $10^{-4}\,bar$.

Table 2: Reaction kinetics of [43]. This model is denoted as "GOS". All concentrations are in mol/m^3 and pressures are in bar.

Reaction	Rate of Reaction $mol\,/\,(m^2.sec)$
Oxidation	$r_1 = 23.06\,\exp\left(\frac{-100320}{RT}\right) C_{\mathrm{CH_4}} C_{\mathrm{O_2}}$
Reforming	$r_2 = 76.16\,\exp\left(\frac{-114120}{RT}\right) C_{\mathrm{CH_4}} C_{\mathrm{H_2O}} \left(1 - \prod_i^{n_{sp}} p_i^{\nu_{i,2}} / K_{eq,2}\right)$
W. G. Shift	$r_3 = 0.412\,\exp\left(\frac{-38130}{RT}\right) C_{\mathrm{CO}} C_{\mathrm{H_2O}} \left(1 - \prod_i^{n_{sp}} p_i^{\nu_{i,3}} / K_{eq,3}\right)$

Table 3: Reaction kinetics represented as "NK" in [26]. Oxidation kinetics is from [108] and reforming kinetics is from [83]. All pressures are in bar.

Reaction	Rate of Reaction $mol\,/\,(m^2.sec)$
Oxidation	$r_1 = 157.7\,\exp\left(\frac{-75500}{RT}\right) \frac{p_{\mathrm{CH_4}} p_{\mathrm{O_2}}}{(1 + K_{\mathrm{CH_4}} p_{\mathrm{CH_4}} + K_{\mathrm{O_2}} p_{\mathrm{O_2}})}$
	$K_{\mathrm{CH_4}} = 2.28 \times 10^{-3}\,\exp\left(\frac{25100}{RT}\right)$
	$K_{\mathrm{O_2}} = 6.41 \times 10^{-4}\,\exp\left(\frac{45900}{RT}\right)$
Reforming	$r_2 = 72.8\,\exp\left(\frac{-106900}{RT}\right) \frac{p_{\mathrm{CH_4}} - p_{\mathrm{CO}} p_{\mathrm{H_2}}^3 / (K_{eq,2} p_{\mathrm{H_2O}})}{(K_{sat} + p_{\mathrm{H_2O}})^{0.596}}$
W. G. Shift	$r_3 = 0.068\,\exp\left(\frac{-54500}{RT}\right) \left(p_{\mathrm{CO}} - \frac{p_{\mathrm{CO_2}} p_{\mathrm{H_2}}}{K_{eq,3}\,(K_{sat} + p_{\mathrm{H_2O}})}\right)$

[1] Available from http://www.reactiondesign.com/

2.2.4 Solution Method and Validation

The control volume method with a non-uniform staggered grid was used to discretize the model equations in the spatial domain to obtain a system of differential algebraic equations (DAE). A standard DAE solver called DASPK [20] was used to integrate the resulting DAE system. DASPK uses a fully implicit fifth order backward difference formula (BDF) with adaptive step size to solve the DAE system. The velocities were computed at grid boundaries and the other variables were computed at nodes (i.e. grid centers). The grid was chosen to be symmetric about the center of the reactor, with the grid boundaries lying at $\frac{l}{2} \times \left(\frac{i}{\text{int}(N/2)}\right)^{\gamma}$, for $i = 1$ to $N/2$, starting at either ends of the reactor. For the value of $\gamma = 1.0$, a uniformly-spaced grid is obtained. Grid independence study was performed with γ value ranging between 1.0 and 2.0, and the optimum values of $\gamma = 1.2$ and $N = 175$ were obtained for good accuracy and a reasonable speed of computation.

During simulations, absolute and relative errors of 10^{-5} were employed for all variables. The consistency of the converged results was checked by computing the residuals of the mass and energy balance equations and verifying that they were less than the desired error tolerance. Prior to using the code for simulation of this system, it was tested against standard examples. Specifically, a standard heat conduction problem and a standard heat transfer problem with first order reaction (with $T_g = T_s$) were solved by making minimum changes to the codes. Satisfactory agreement was obtained between simulations and analytical solutions for standard problem.

2.3 Time Scale Analysis

Consider a small volume element within the reactor. Figure 1 shows the interactions between various individual processes occurring within this element. The material and energy enter and exit this element by advection; transfer of species occurs from the bulk gas to the catalyst surface by convection and vice versa; several reactions occur on the surface of the catalyst; these reactions result in a net release or consumption of heat; and there is also heat exchange between the solid and the bulk gas. Each of these processes is associated with its own time scale. In addition, the thermal capacity of the reactor is also associated

with an 'inertial' time scale of thermal relaxation. A time scale is defined as "the time required for an individual process to occur assuming that all other processes do not affect the reactor performance."

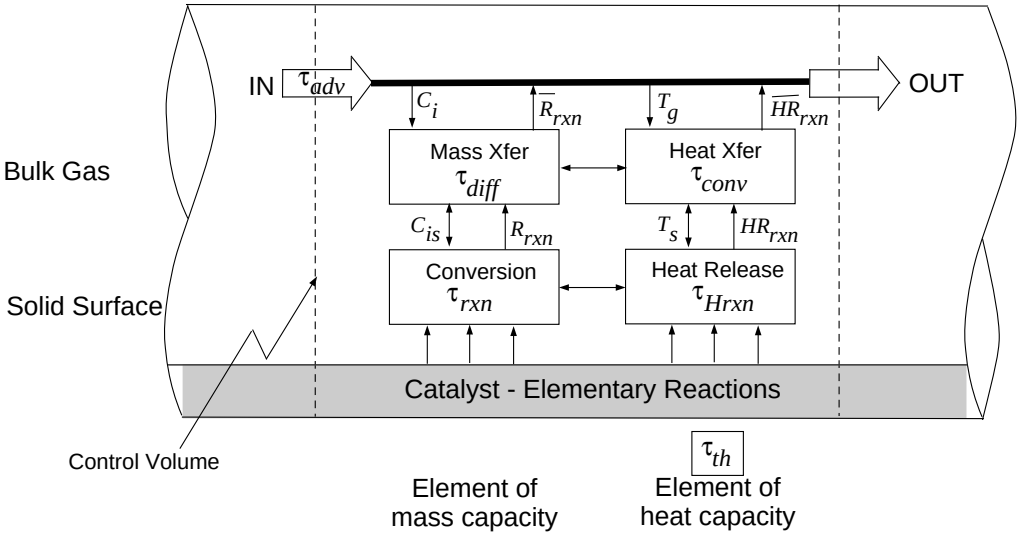

Figure 1: Interaction of various individual processes within the reactor and their time scales

Table 4: Time scales of various processes within the microreactor

Process	Time scale	Value (sec)
Advection	$\tau_{ad} = l/v$	0.013
Diffusion	$\tau_{diff} = (k_g \underline{a})^{-1}$	0.00014
Oxidation	τ_{ox}, numerical	0.002
Reforming	τ_{ref}, numerical	0.008
W. G. Shift	τ_{wgs}, numerical	0.006
Reaction Heat	Eq. (15)	3.2
Thermal Inertia	τ_{th}, numerical	168

The time scales of various processes are listed in Table 4. Naturally, the time scales are computed at the locations where their effect is the highest; *i.e.* at or near the reactor hot spot where the reaction rates are the highest. These conditions were obtained from the simulation results for the nominal case, presented in the next section. The reaction time

scale was numerically computed as the time required for the reaction to reach equilibrium (or complete conversion for oxidation), starting at concentrations and temperature existing in the bulk gas. The time scale of heat release was computed as shown in [72] by linearizing the reaction heat source term:

$$\frac{dT_s}{dt} = \frac{a}{\rho_s c_s} \sum_{j=1}^{n_{rxn}} [-\Delta H_j] r_j(C, T_s)$$

resulting

$$\tau_{Hrxn} = \frac{\rho_s c_s}{a} \left[\sum_{j=1}^{n_{rxn}} \left[\frac{E_j}{RT_s^2} \right] [-\Delta H_j] r_j(C, T_s) \right]^{-1} \tag{15}$$

Thus, the time scale of heat source is the time required for the solid temperature to increase by the value RT_s^2/E due to heat release. It is the combined effect due to various reactions concurrently taking place within the system.

The time scale of thermal relaxation was also calculated numerically. The system was simulated assuming no reaction, radiation or heat losses. The energy balance equations for the solid and gas phase were solved simultaneously with $T_s(z, t = 0) = 1173\,K$ and $T_g(0, t) = 300\,K$. The τ_{th} is the time taken for the temperature of an element in the interior of the reactor to drop from $T_s = 1130\,K$ (5% of the temperature difference) to $343\,K$ (95% of the temperature difference). Note that τ_{th} is the response time of a single volume element in the reactor, which is much less than the time it takes for the reactor to reach steady state.

The simple time scale analysis can give some very important information about the system. First, the time scale of diffusion is smaller than that of the reactions, meaning that the reactions occur at their intrinsic rates. Further, as the time scale for advection is higher than that of all the reactions, the reactions reach their equilibrium (quasi-steady state), and the overall thermal effects within the system are critical in defining the reactor operation in UD or RF modes.

2.4 Analysis of Reaction Kinetics

In this section, the simulation of the reactor in unidirectional (UD) and reverse-flow (RF) operation using the GOS and NK models is compared. The axial variations in the velocity are incorporated using the constant pressure assumption as discussed in section 2.2.2. The

operating conditions for the simulation of the "baseline case" are taken from those reported in the experiments [59], and are summarized in Table 1.

2.4.1 Unidirectional and Reverse-Flow operation: GOS model

We first consider the comparison of UD and RF operation of the adiabatic reactor for the GOS model. Figure 2 shows the temperature profiles in the reactor. In the UD operation, the temperature reaches a maximum in the middle of the reactor, where highly exothermic oxidation dominates. A stable, autothermal steady state is obtained for UD operation of the reactor. For feed ratio of $CH_4 : O_2 = 1 : 1$, inlet velocity of $v_0 = 1.68\, m/s$ and inlet gas temperature $T_{g0} = 300\, K$, the maximum temperature reached was $2008\, K$ and the hydrogen yield at steady state was 71.1%.

In RF operation with 5 seconds flow reversal time (the period used in experiments), the temperature peak is attained closer to the reactor ends (Figure 2). The central section of the reactor is maintained at a more uniform temperature, with a sharp drop in the temperature at the reactor end. The widening of the high temperature region also results in a lower temperature peak. The sharp temperature drop at the end of the reactor favors water gas shift reaction, resulting in an improvement in hydrogen yield. Figure 3 shows the mole fractions of all the species in RF operation after attainment of a periodic steady state, just prior to switching of the flow direction. Table 5 shows the comparison between UD and RF operations. The GOS model predicts a 2.3% improvement in the hydrogen yield and a $120\, K$ decrease in the peak temperature for $CH_4 : O_2 = 1 : 1$ and a 4.2% increase in the yield for $CH_4 : O_2 = 0.7 : 1$. These values are consistent with those reported in the experiments [59].

2.4.2 Simulation using NK Model

The effect of kinetic rate equations was considered by simulation of indirect partial oxidation using the NK model (see Table 3). The simulations results using this model are compared with those from the GOS model. The results of UD and RF operations are shown in Table 5. The maximum temperature reached in this case was well over $2300\, K$. This

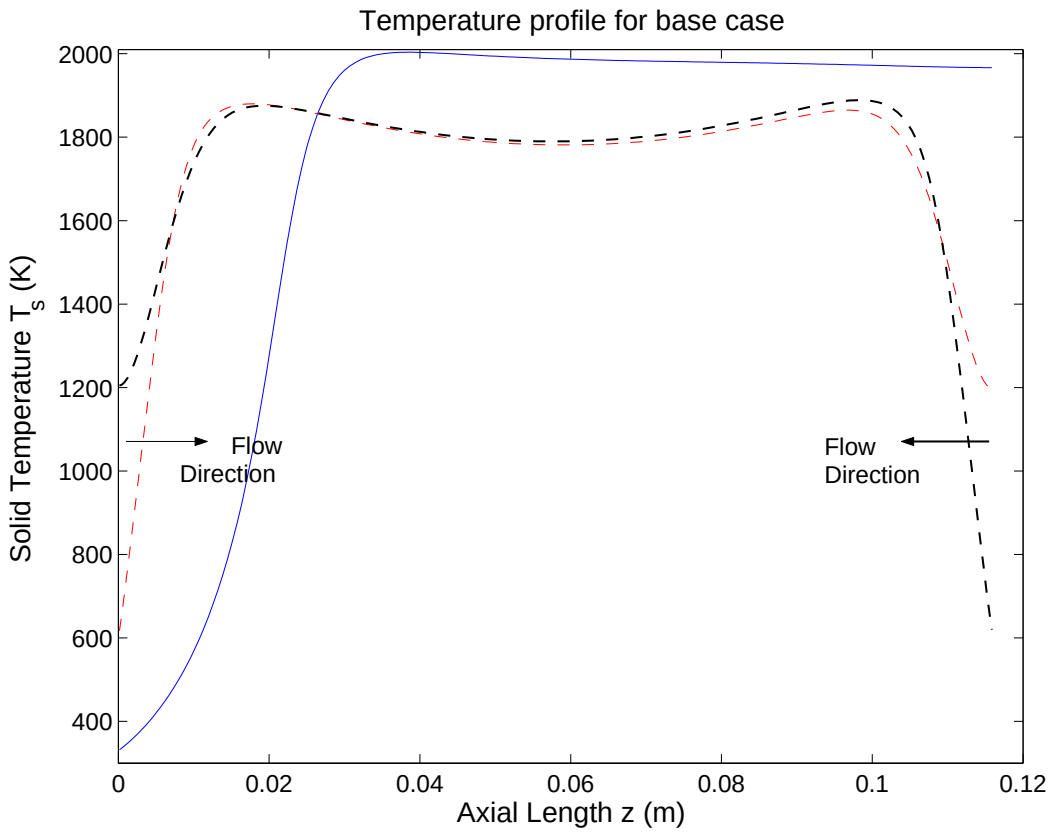

Figure 2: Temperature profiles in UD (at steady state) and in RF (just prior to flow reversal) for the GOS model. The temperature profile is more uniform and the peak lower in RF operation.

is significantly higher than that predicted by the GOS model due to the low rates of endothermic reforming reactions in the NK model. Increasing the rate of reforming reactions (to assess the sensitivity of the results) resulted in disappearance of the autothermal steady state in UD operation. Moreover, RF operation did not result in an increase in hydrogen yield over UD mode, even for $CH_4 : O_2 = 0.7 : 1$. Thus, this model was unable to predict the experimentally observed behavior of the microreactor reported in [59]. Varying the reaction rates within the order of magnitude (by a factor of 10) did not predict the observed behavior either. This is because the rate of water gas shift reaction is significantly lower in this model. The shift reaction is thermodynamically favored at low temperatures that exist in RF operation at the reactor exit. The exit temperatures in the NK model are still quite high. The rate of water gas shift reaction is significantly lower than in the GOS model.

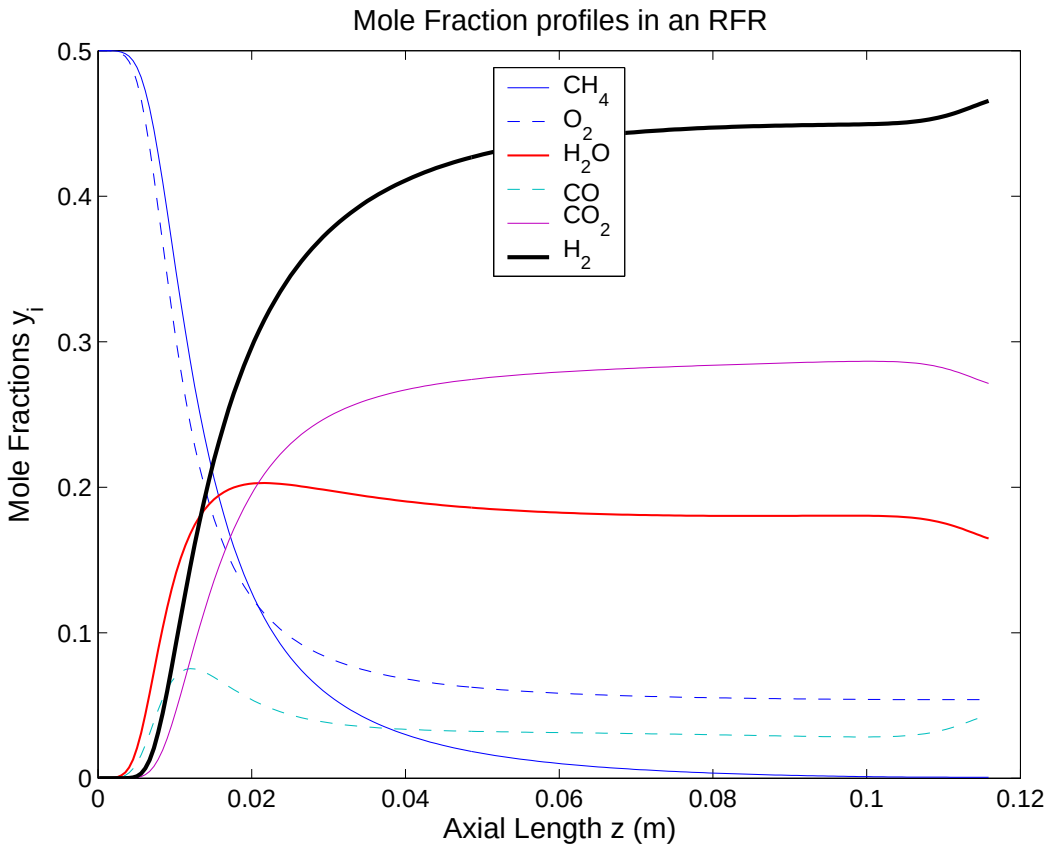

Figure 3: Mole fractions of various species in gas phase just prior to flow reversal in RF operation with 5 *sec* flow reversal time.

Hence, no difference between UD and RF operations was observed for the NK model.

2.5 Effect of Radiation

At high temperatures such as those observed in this system, heat transfer by radiation becomes comparable to (or even dominant over) heat transfer by conduction or convection, due to the fourth order dependence of the radiative heat flux on the temperature. In addition to this nonlinear dependence on temperature, radiation heat transfer in a non-participating medium is a long range phenomenon and has complex dependence on properties of the medium and the reactor surface. Thus, incorporating radiation involves solving highly nonlinear integro-differential equations. As a result, radiation has often been ignored in the literature dealing with partial oxidation / reforming.

With a conservative estimate of solid temperature of 1500 K, the radiation to conduction

Table 5: Reactor performance for experimental conditions [59] with GOS and NK kinetics. The experimentally observed trends are matched by the GOS model but not by the NK model.

Model	$CH_4 : O_2$ ratio	Unidirectional		Reverse-Flow		
		$T_{s,max}$	$\%H_2$ yield	$T_{s,max}$	$T_{s,out}$	$\%H_2$ yield
GOS	0.7 : 1	2160	57.3	2023	1280	61.5
GOS	1 : 1	2008	71.1	1880	1191	73.4
NK	0.7 : 1	2561	57.8	2479	1648	58.2
NK	1 : 1	2308	75.3	2207	1421	74.9

heat transfer parameter $N_{rad} = \epsilon\sigma T^3 d/\lambda$ [77] takes the value of 0.65, indicating that the radiative heat transfer is comparable to solid conduction. In this section, we evaluate the effects of radiation using the net radiosity method [77] to compute the radiation flux within the reactor. We make an assumption that the walls are diffuse gray emitters and reflectors of radiation, and that the medium is essentially transparent (i.e. non-participating) due to the small channel diameters. All the results reported here are for reactor wall emissivity of $\epsilon = 0.8$.

For channels with high aspect ratio (in our system $l/d = 232$), the finite-difference version of the net radiosity equation can be used instead of the Fredholm integral equation [15]. The finite-difference version ties up well with the spatial discretization used in solving the other conservation equations. The outgoing (leaving the wall) radiation flux at any point z_n within the reactor channel is given by

$$\begin{aligned} q_r(z_n) \quad = \quad & \sum_{m=0}^{N} (1-\epsilon) q_r(z_m) F_{n-m} + \\ & \epsilon \cdot \left[E_b(z_n) - \sum_{m=0}^{N} E_b(z_m) F_{n-m} - \sigma T_{in}^4 F_{re}(z_n) - \sigma T_{out}^4 F_{re}(l - z_n) \right] \end{aligned} \tag{16}$$

where F_{n-m} is ring-to-ring view factor and F_{re} is ring-to-end view factor. The finite view factors are derived using a procedure similar to that presented by Rankin et al. [90] and is given in Appendix A.

Figure 4 compares the solid temperatures in the system with and without radiation

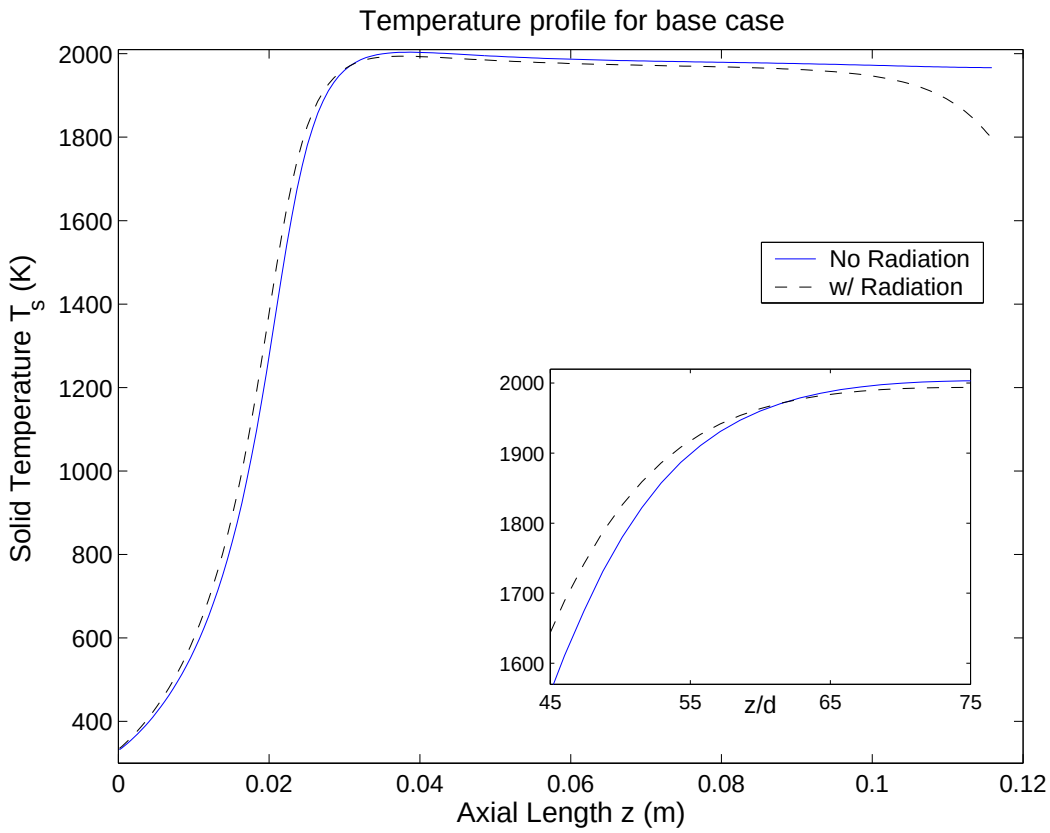

Figure 4: Effect of radiation on solid temperature. The temperature profile shifts upstream by 2 diameters (inset: zoomed-in at temperature peak, dimensionless length z/d is the abscissa). Radiation loss at the reactor end causes a drop in temperature. Concentration profiles (not shown) are almost same in both cases.

accounted for. For the reactor with high aspect ratio, the effect of radiation is not significant. The figure shows that the peak temperature is shifted towards the reactor inlet by about 2 diameters, which for the simulated system is less than 1% of the reactor length. The inset shows a zoom-in plot near the temperature peak. The abscissa for this plot is normalized length, which clearly shows an upstream shift of about 2 diameters in T_s. These results are consistent with the published literature [15, 38], where the temperature curve shifts upstream by 1 to 2 diameters. Moreover, the heat loss due to radiation at the reactor outlet causes a drop in the temperature at the outlet.

Although the total black body emissive power $E_b(z) = \sigma T_s(z)^4$ is high (same order of magnitude as convection flux), radiation does not seem to have a marked effect. The reason

for such a behavior can be easily seen in the view factor plot 5. The ring-ring view factor, which determines what fraction of total radiation emitted by a ring element is incident on another surface ring element, diminishes to 0 rapidly near the reactor hot spot. Almost 70% of the radiation emitted by a differential ring element of the reactor is incident on itself. Hence, the computed radiation flux (Figure 6) is two orders of magnitude less than the heat release by reaction and the convective flux.

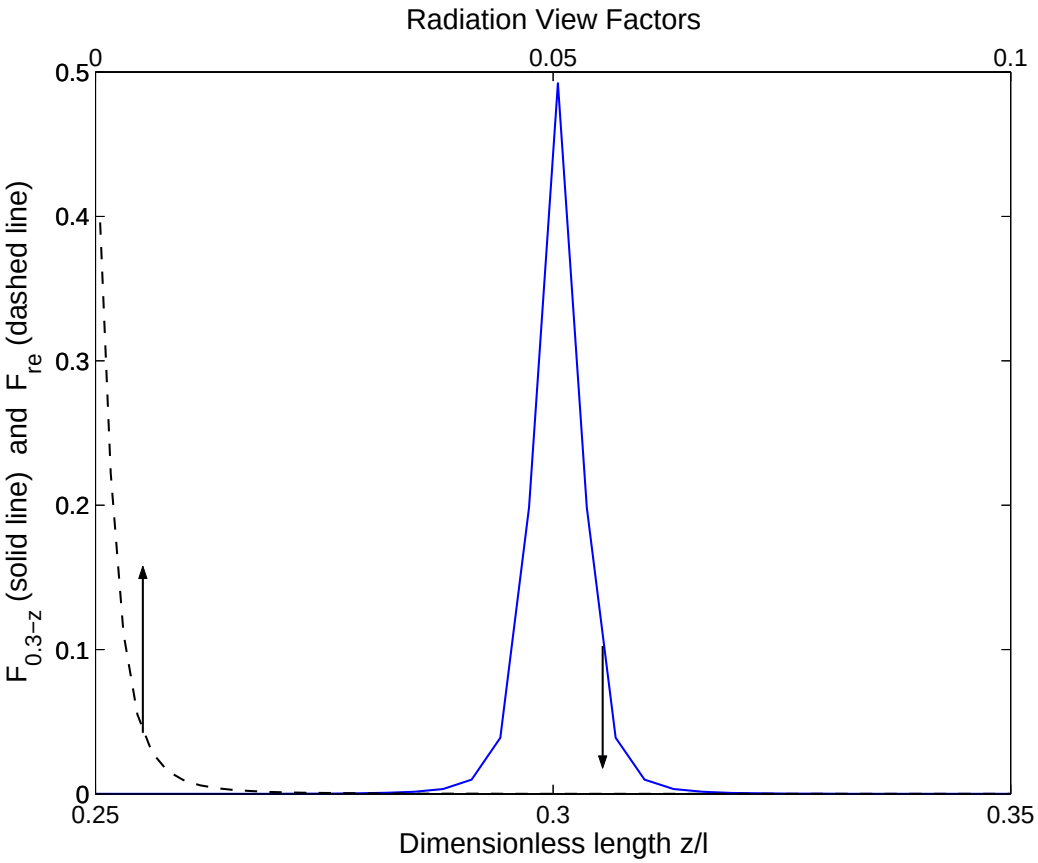

Figure 5: Ring-ring (solid line, bottom abscissa) and ring-end (dashed line, top abscissa) view factors as a function of the dimensionless axial coordinate. The view factors diminish rapidly with the axial distance.

Finally, the effect of radiation on the shorter (5 cm long) reactor was also studied. The reactor length of 5 cm was chosen to give an aspect ratio of $l/d = 100$. The temperature and mole fractions of hydrogen obtained in this reactor are shown in Fig. 7. Again, there is a 2-diameter shift in the temperature profile. This is in agreement with Fu et al. [38], for the aspect ratio $l/d < 10$, the effect of radiation was significant at temperatures lower than

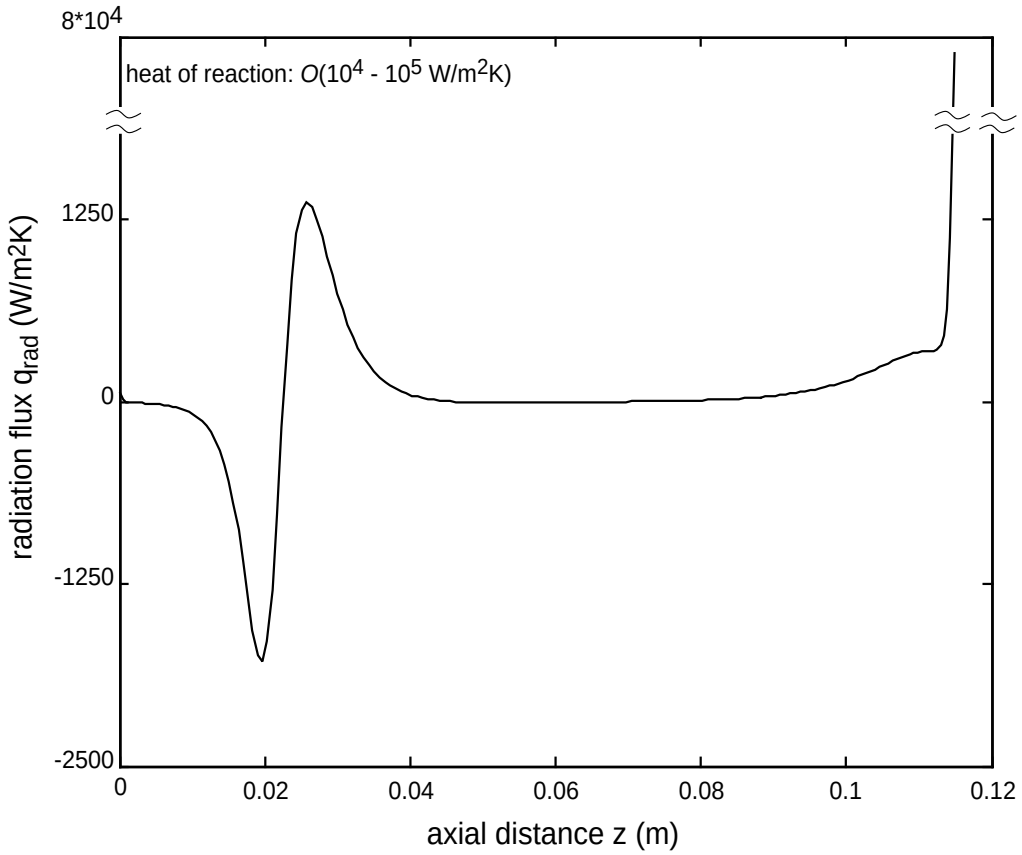

Figure 6: Radiation flux at steady state within the reactor. Except at the reactor end, radiation flux is 1 to 2 orders of magnitude lower than convection or reaction heat fluxes.

in our system. However, the temperature drop at the reactor end causes a significant drop in hydrogen yield. The decrease in the hydrogen yield was about $2 - 3\%$. We also found that changing the emissivity resulted in temperature and concentration profiles similar to those obtained with $\epsilon = 0.8$. Thus, one can use the black body assumption to reduce computational burden in calculating the radiation flux, without increasing the error by much. All these results are consistent with those reported in the literature [15, 38].

2.6 Conclusions

A 1-D model for simulation of autothermal partial oxidation of methane in a microreactor with unidirectional (UD) and reverse-flow (RF) modes of operation was developed. Two different kinetic models for methane oxidation and reforming from the literature were

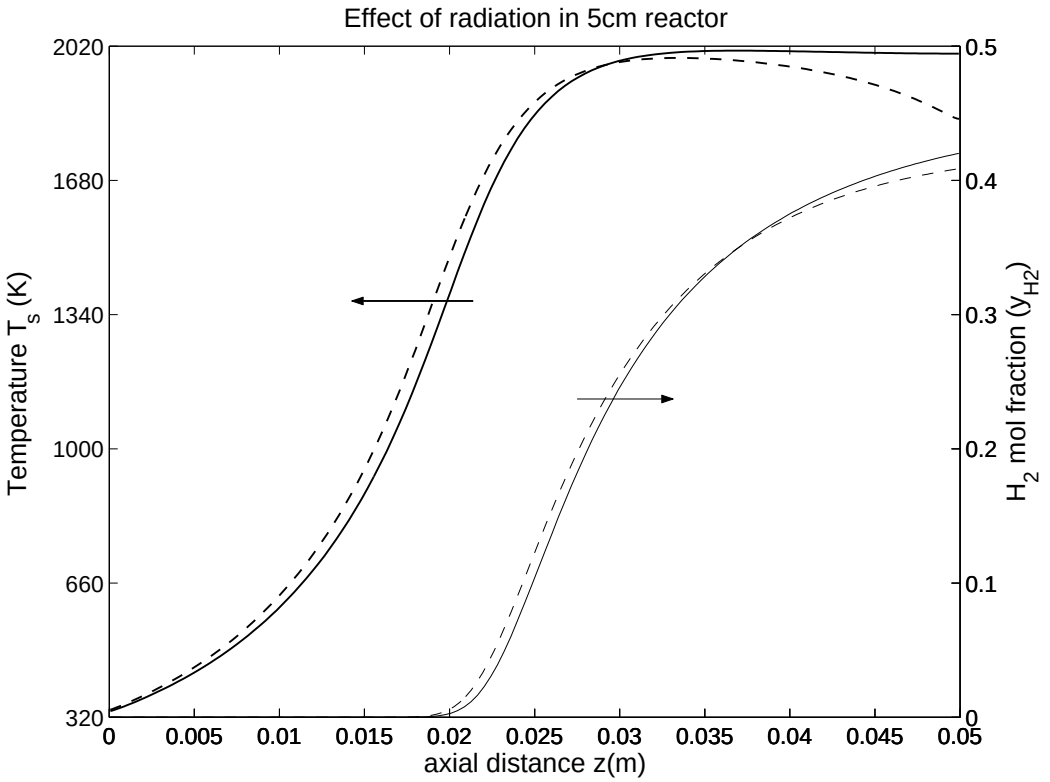

Figure 7: Effect of radiation in a $5cm$ reactor. Thick lines: T_s, thin lines: y_{H_2}; solid line: without radiation, dashed line: with radiation. Temperature drop at the reactor end causes a decrease in H_2 yield.

compared; an analysis of time scales of individual processes occurring in the reactor was presented; and the effect of radiation heat transfer on reactor performance was analyzed. Specific results of this work include:

- The reaction kinetics presented by Gosiewski et al. [43] (GOS model) and de Smet et al. [26] (NK model) were compared. The GOS model was able to reproduce the experimental results reported by Kikas et al. [59]. In the best case, a 4.2% improvement in hydrogen yield in RF operation was predicted. The NK model, on the other hand, did not show any improvement in the reactor performance in RF operation.

- An analysis of time scales of individual processes occurring within the reactor was presented. The time scale of diffusion was lower than that of reactions. Hence, the reactions occurred at their intrinsic rates.

- Based on the time scales, the species concentrations were found to be under quasi-steady state and the overall thermal dynamics of the reactor were critical in defining the reactor operation in UD or RF modes.

- Radiation heat transfer did not have a significant impact on the performance of the longer reactor. Consistent with the published literature [15, 38], radiation resulted in 1 to 2 diameter shift in the temperature profile and a sharp drop in the temperature at the reactor exit. However, due to the large aspect ratio ($l/d = 232$) in the reactor, the overall effect was insignificant.

- In contrast to this, the hydrogen yield in a shorter reactor (aspect ratio $l/d = 100$) with radiation included was about 2% lower than that without radiation.

CHAPTER III

SIMULATION AND ANALYSIS OF THE REVERSE-FLOW OPERATION

3.1 Introduction

Forced unsteady state operation (see [74] for an excellent review) of chemical reactors and separation processes often lead to an improved performance over the steady state operation. For example, Horn and Lim [50] used a variational approach to determine the optimal periodic operation of a CSTR; Horn [49] showed an improvement in efficiency of a periodically operated separation processes; Sterman and Ydstie [100, 101] used the so-called 'pi-criterion' to analyze the feasibility of a periodic operation; Bailey [6] employed convex set theory for optimization in periodic control; Eigenberger and Nieken [30] examined combustion of volatile organics at low concentrations in reverse-flow reactors (RFRs), while Haynes et al. [45] proposed a procedure for the design of RFRs. The last two examples in this list involve operating a reactor in a reverse-flow (RF) mode through periodic switching of the input and output ports resulting in a reversal of the flow direction. Matros and Bunimovich [75] provide a comprehensive review of theoretical and experimental developments in RFRs. The RF operation often leads to transient patterns in a catalytic system which are not found in steady state operation. The opportunity for improvement in the performance of an RFR occurs due to dynamic properties on the catalyst and/or due to the dynamic properties of the whole reactor [73].

In this work, partial oxidation of methane in a reverse-flow microreactor is considered. Previous work on RF operation of methane partial oxidation has mostly focussed on fixed bed reactors or catalytic monoliths. Blanks et al. [14] were amongst the first to provide experimental and simulation study on a pilot scale RFR for methane partial oxidation. de Groote et al. [25] showed that a thermal wave is formed in a fixed-bed partial oxidation

reactor, which travels along the length of the reactor. They showed that this travelling wave in an RFR goes through three different stages in each semi-cycle: the wave development stage, wave widening stage and the wave propagation stage. Gosiewski and coworkers compared the performance of an RFR [43] with a unidirectional reactor with periodic feed cycling [42]. Fissore et al. [34] used their model for simulation of periodic changing of feed location in a three-reactor network to vary the sequence of reactors.

Modelling of methane partial oxidation for autothermal operation of a microreactor running in unidirectional (UD) and reverse-flow (RF) modes and an analysis of time scales of individual processes occurring within the reactor was presented in chapter 2. Using the reaction kinetics adapted from Gosiewski et al. [43], we were able to reproduce the experimental result of Kikas et al. [59] that the RF operation of the reactor provides higher hydrogen yield and lower maximum temperature as compared to the UD operation of the microreactor. In this chapter, we use the time-scale analysis to obtain the origins of the observed improvement in the performance with RF operation. As the reaction and diffusion time scales are very fast, species concentration reach a quasi-steady state. Therefore, there are the dynamic thermal properties of the entire reactor that are exploited to obtain process intensification in RF operation of the microreactor. Specifically, the RF operation may result in one or more of the following advantages [75]:

- Creating conditions that are thermodynamically favorable for reversible reactions
- Efficient energy utilization resulting in a more uniform temperature pattern, consequently resulting in better performance at a smaller reactor size and lower average reactor temperature
- Using the RFR as a regenerative heat exchanger that allows the autothermal operation of weakly exothermic processes

These three benefits are highlighted in Fig. 8. Figure 8-(a,b,c) show the schematic of temperature profiles for three different situations in UD operation, while Fig. 8-(d) shows the corresponding temperature profile for RF operation. Figure 8-(a) shows the case wherein autothermal UD state is obtained and the reactor is long enough for the desired reaction

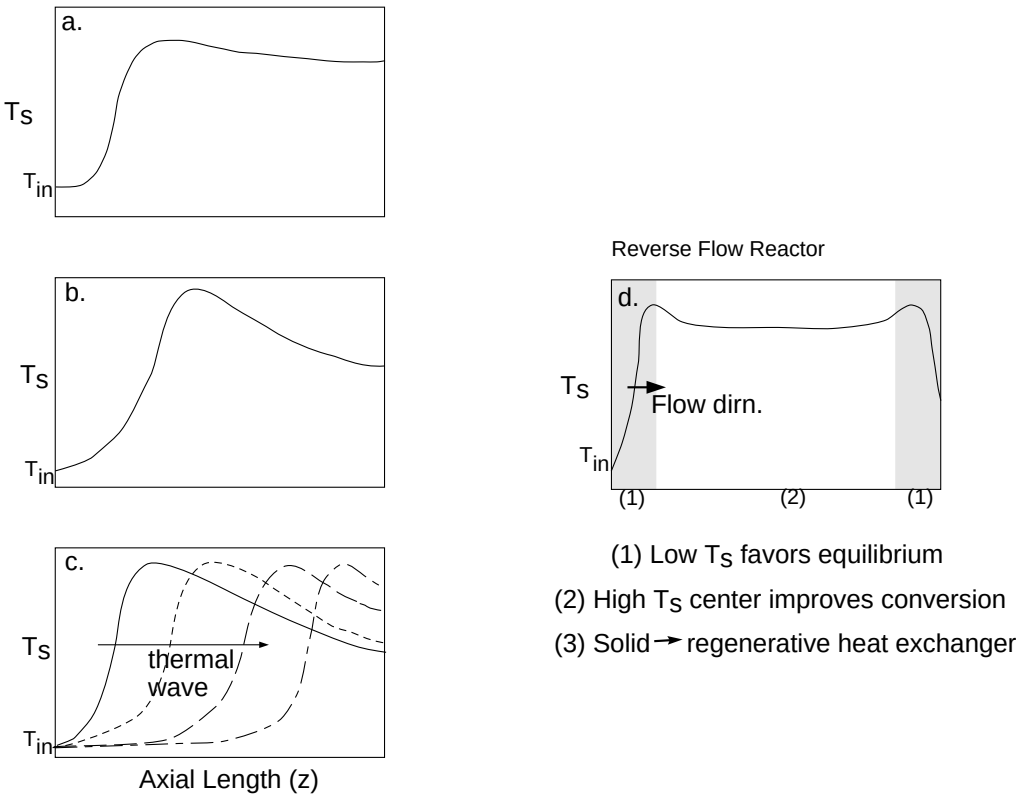

Figure 8: Temperature profiles in UD operation for three possible conditions: (a) reactions proceed to equilibrium, (b) poor utilization of reactor, (c) thermal wave travels and exits the reactor. In RF operation (d), there is an extended region (2) of high temperature where reactions reach equilibrium flanged between the shaded regions (1) of thermodynamically favorable conditions at reactor ends.

to reach equilibrium. The gas exits the reactor at a relatively high temperature, which is thermodynamically unfavorable for an exothermic reaction. In RF operation, however, the sharp drop in the temperature at the reactor exit provides a thermodynamically favorable condition, shifting the equilibrium towards the desired product. A similar trend is also observed in an RFR with high temperature feed for an endothermic reversible reaction. The second case (Fig. 8-(b)) is usually found when endothermic and exothermic reactions are coupled and the reactions do not reach equilibrium within the reactor. The temperature decreases on the either sides of the hot spot in UD operation, resulting in lower rates of reactions and a poor utilization of the reactor. In an RFR, flanged between the two hot spots near the reactor ends is an extended region of high temperature. The rates of endothermic

reactions are high in this region. Consequently, the conversion and yield of the desired product are also higher. The third and perhaps the most widely studied example is that of a slightly exothermic high temperature reaction, which is unable to to provide sufficient heat to maintain UD steady state. In such a situation, a travelling wave is usually formed. The RFR is then used as a regenerative heat exchanger, trapping the thermal wave within the reactor by flow reversal before the reaction front exits the reactor.

The rest of the chapter is organized as follows. The simulation results are presented in the section 3.2: the effects of changing the $CH_4 : O_2$ feed ratio, presence of water in the feed, and the variations in velocity or gas temperature at the inlet are studied. We also consider the effect of reactor length and the reactor heat losses on the reactor performance. The analysis of the RF operation is presented in section 3.3. All three circumstances for process intensification mentioned in the previous paragraph were observed for different conditions in the simulated reactor and are discussed in this section. The effect of varying the switching time is studied and guidelines for optimal reactor operation based on the scale analysis are presented. Finally, the key findings are summarized in the concluding section.

3.2 Simulation Results

3.2.1 Adiabatic reactor: Base case

The "base case" refers to simulations of the adiabatic reactor for the parameters used in the experiments [59]. These conditions are summarized in Table 1 and simulation results presented in Table 5. The reactor consists of four channels 500 microns in diameter and 11.6 cm long. In this section, simulation results for varying input conditions for the reactor in unidirectional (UD) and reverse-flow (RF) operation are discussed.

3.2.1.1 Inlet feed concentrations

The effect of varying feed ratios $CH_4 : O_2$ is shown in Table 6 and in Fig. 9. As the feed ratio $CH_4 : O_2$ is increased, the selectivity to hydrogen increased, resulting in an increase in hydrogen yield. This happened until $CH_4 : O_2 = 1.15 : 1$. Beyond this point, the hydrogen yield decreased again and an optimal feed ratio was found to be $CH_4 : O_2 = 1.15 : 1$. This compares favorably with the experimentally reported optimal ratio of $\approx 0.9 : 1$ [59]. This

is a departure from the published literature [43, 25], where a ratio closer to the partial oxidation ($CH_4 + 0.5O_2 \rightleftharpoons CO + 2H_2$) stoichiometric ratio of 2 : 1 is generally used. In the best case, a 4% improvement in the yield in RF over UD operation was observed. At $CH_4 : O_2$ feed ratios of 1.25 : 1 and higher, the model predicted lower hydrogen yield in RF operation compared to the UD operation. A slowly creeping thermal wave was observed for $CH_4 : O_2 = 1.5 : 1$. Due to low speed of wave propagation within the reactor, stable UD operation was possible for 2–3 hours before the reactor quenched. Finally, with water present in the feed, autothermal UD operation was not obtained. In contrast, at lower amounts of water in the feed, autothermal RF operation was possible although there was a drop in the hydrogen yield. When the amount of water was increased further, it resulted in disappearance of autothermal operation even in the RF operation of the reactor. These results are consistent with those observed experimentally [59].

Table 6: Reactor performance for various inlet feed ratios. A travelling thermal wave is formed at feed ratio of 1.5 : 1 and higher. *: The wave creeps through the reactor, and the reported hydrogen yield is obtained for over 2 hours of operation. *tw* : travelling wave, no UD autothermal state is observed.

Case	$CH_4 : O_2$ ratio	Unidirectional		Reverse-Flow		
		$T_{s,max}$	$\%H_2$ yield	$T_{s,max}$	$T_{s,out}$	$\%H_2$ yield
1	0.7 : 1	2160	57.3	2023	1280	61.5
2	1 : 1	2008	71.1	1880	1191	73.4
3	1.15 : 1	1951	77.5	1816	1132	77.4
4	1.25 : 1	1914	77.0	1779	1093	76.6
5	1.5 : 1	1841	74.4*	1703	1007	72.6
6	1.75 : 1	tw	tw	1660	939	65.2
7	2 : 1	tw	tw	1608	890	59.2
8	$CH_4 : O_2 = 1 : 1$ $CH_4 : H_2O = 3 : 1$	tw	tw	1734	861	67.9

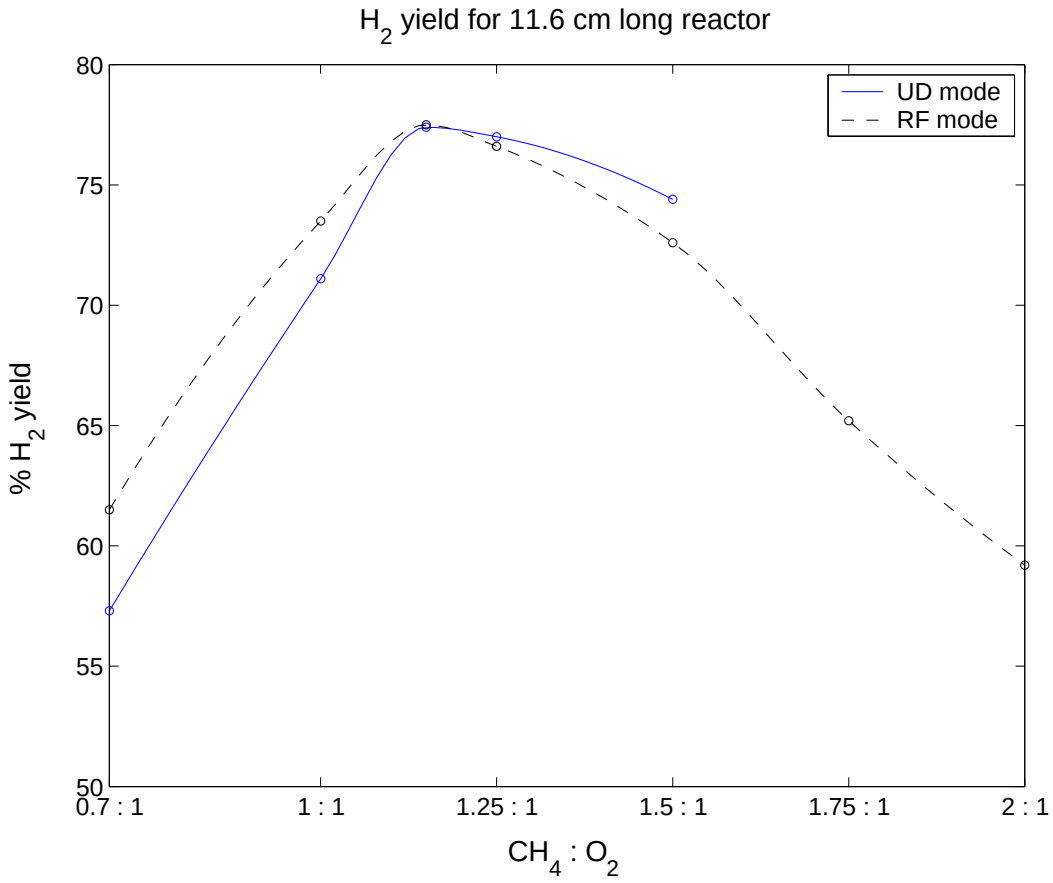

Figure 9: Comparison of UD (—) and RF (– –) operation for various $CH_4 : O_2$ ratios. UD steady state cannot be obtained for feed ratios of 1.5 : 1 or greater. Also see Table 6

3.2.1.2 Influence of the inlet velocity

Figure 10 shows the variation in hydrogen yield in UD and RF operation with the inlet velocity for constant feed ratio of $CH_4 : O_2 = 1 : 1$ and inlet gas temperature $T_{g0} = 300\,K$. Changing the velocity had substantial effect on the reactor performance. In UD operation, hydrogen yield showed almost a linear dependence on the inlet velocity. Increasing the velocity to $1.8\,m/s$ resulted in the formation of a thermal wave and the autothermal UD steady state disappeared. When the inlet velocity was increased further, the speed of propagation of the thermal wave increased almost linearly with an increase in the inlet velocity. On the other hand, even when the velocity was decreased to $0.5\,m/s$, the movement of the reaction front opposite to the direction of the flow (flash back) was not observed.

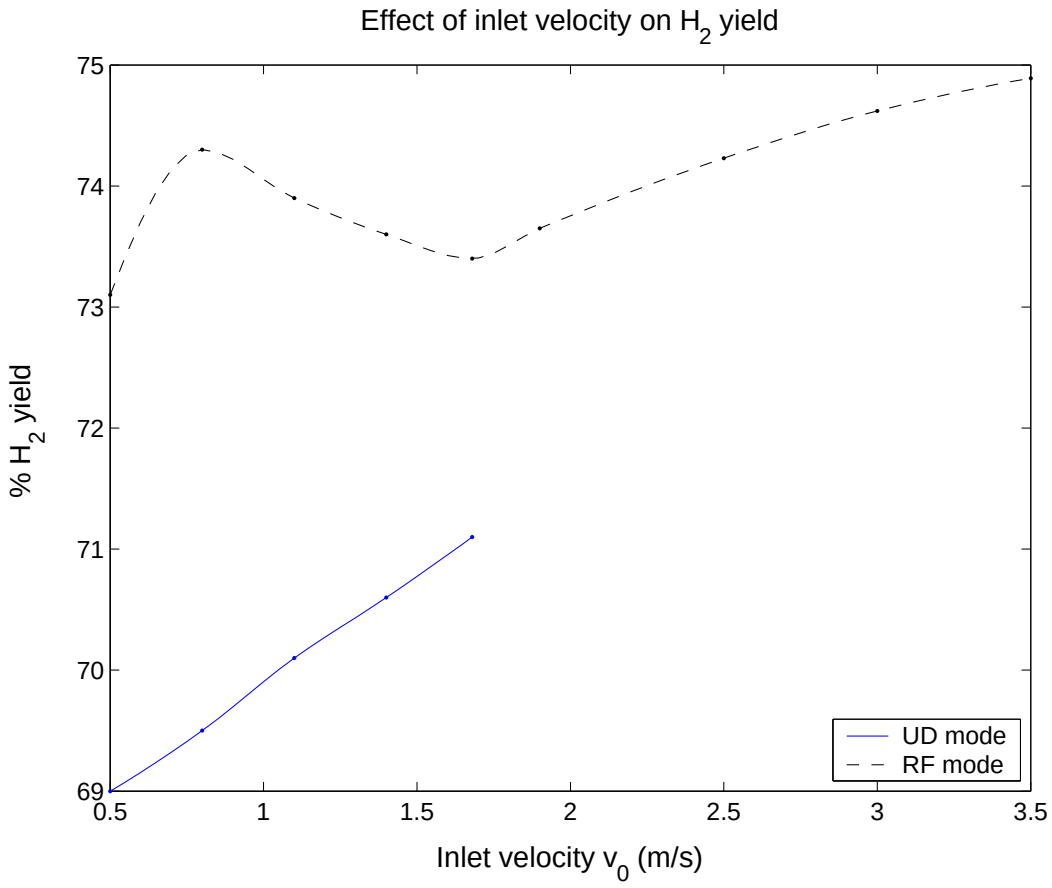

Figure 10: Effect of varying inlet velocity on H_2 yield. Autothermal UD state cannot be obtained for $v_0 = 1.8\,m/s$ or higher.

The effect of inlet velocity on RF operation was quite different. Increasing v_0 resulted in a decrease in hydrogen yield until about $1.8\,m/s$. Recall that this is the point where thermal wave was formed in UD operation. When the inlet velocity was increased beyond this point, the hydrogen yield also increased and finally reached a "saturation" at $v_0 \approx 3.5\,m/s$. The effect of velocity on solid temperature (not shown) in UD and RF operation was more predictable: increasing the inlet velocity resulted in a monotonic increase in the solid temperature due to higher throughput of the reactants through the microreactor. Consequently, for low velocities (eg. $v_0 < 0.75\,m/s$), overall conversion to hydrogen dropped due to lower peak temperatures.

3.2.1.3 Influence of the inlet gas temperature

The effect of varying the inlet gas temperature on UD operation, as seen in Fig. 11 for a constant inlet velocity $v_0 = 1.68\,m/s$ was less significant than that of the inlet velocity. Hydrogen yield decreased slightly with an increase in T_{g0} in both UD and RF operations; however, the hydrogen yield in RF operation decreased more noticeably than the UD operation. Around $T_{g0} = 573K$, the RF operation provided little improvement over UD operation (Fig. 11). In addition to this, increasing the T_{g0} had an effect of stabilizing travelling thermal wave; in fact, when T_{g0} was increased to $573\,K$, an autothermal UD steady state was observed for $CH_4 : O_2 = 2 : 1$ with inlet velocity of $v_0 = 1.68\,m/s$.

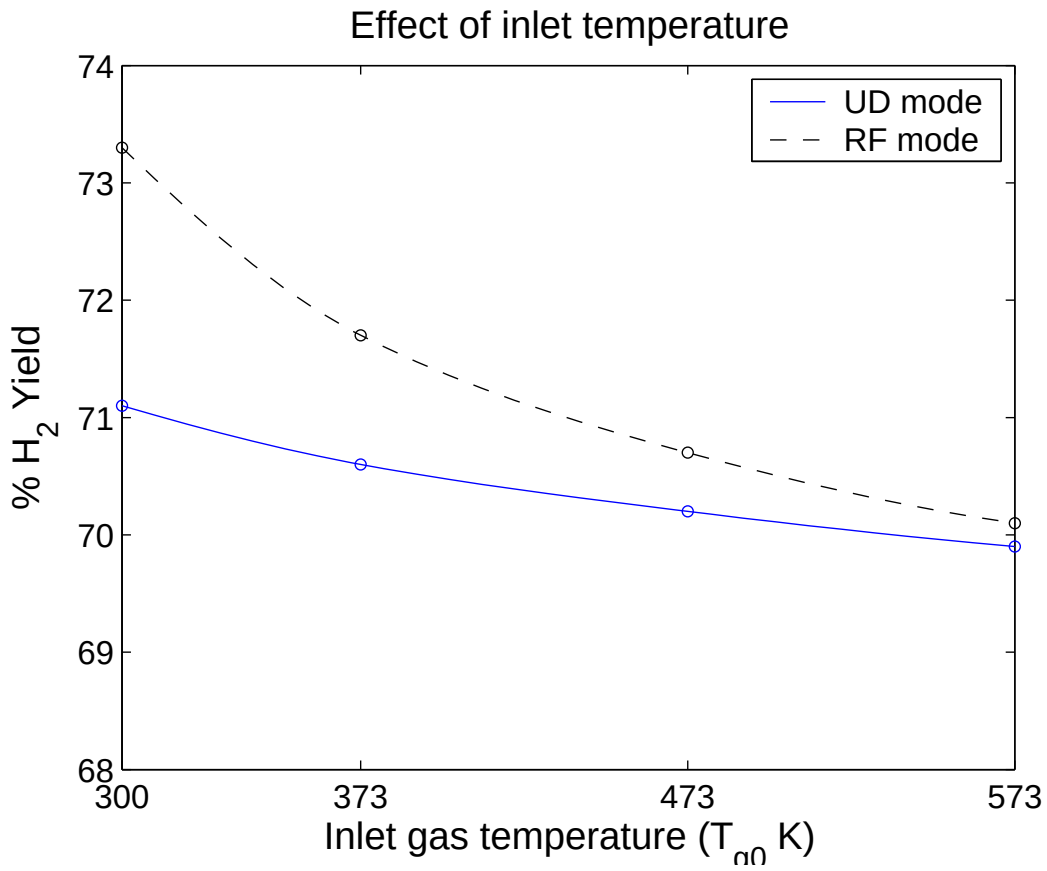

Figure 11: The effect of varying inlet gas temperature (T_{g0}) for $v_0 = 1.68\,m/s$ and $CH_4 : O_2 = 1 : 1$ on hydrogen yield. H_2 yield decreases as the inlet temperature increases.

3.2.2 Effect of Reactor Length

The effect of varying the reactor length was studied by comparing the results obtained for the 11.6 cm-long reactor in the previous section with those for a 5 cm-long reactor. Figure 12 shows the performance of a 5 cm long reactor in UD and RF operation. This reactor length is insufficient for the reactions to attain equilibrium. Hence, the hydrogen yield for UD operation is much lower for the 5 cm reactor compared to the longer reactor. The optimum feed ratio also changed: for UD operation of the shorter reactor, $CH_4 : O_2 \approx 0.9 : 1$ was optimal. What is interesting to note that the improvement in the performance for RF over UD operation was greater in the shorter 5 cm reactor. For the entire range of feed conditions, the RF operation provided much better hydrogen yield than UD operation for this shorter reactor. Unlike the UD operation, the optimum feed ratio for RF operation did not change much from that found in long (base case) reactor (i.e., optimal $CH_4 : O_2 \approx 1.1 : 1$).

3.2.3 Effect of Heat Losses

One of the main concerns in maintaining the autothermal operation of a microreactor is managing heat losses to the surroundings. For the reactor under consideration, the heat transfer coefficient for heat loss to the surroundings, expressed based on the inner channel diameter was estimated to be $0.32 W/m^2K$. Simulations were performed for this case. The overall trends observed were similar to the ones reported for the adiabatic reactor case and are therefore skipped for the sake of brevity.

Next, we considered the case where the system is poorly insulated. In such a situation, the heat transfer coefficient expressed based on the inside channel diameter could be as high as $3.2 W/m^2K$. In this situation, the reactor did not exhibit an autothermal UD operation. However, the autothermal operation can be maintained in RF mode. The pertinent results are given in Table 7. This is a significant result because it shows that the RF operation is a robust method for on-demand generation of hydrogen even in the presence of large heat losses.

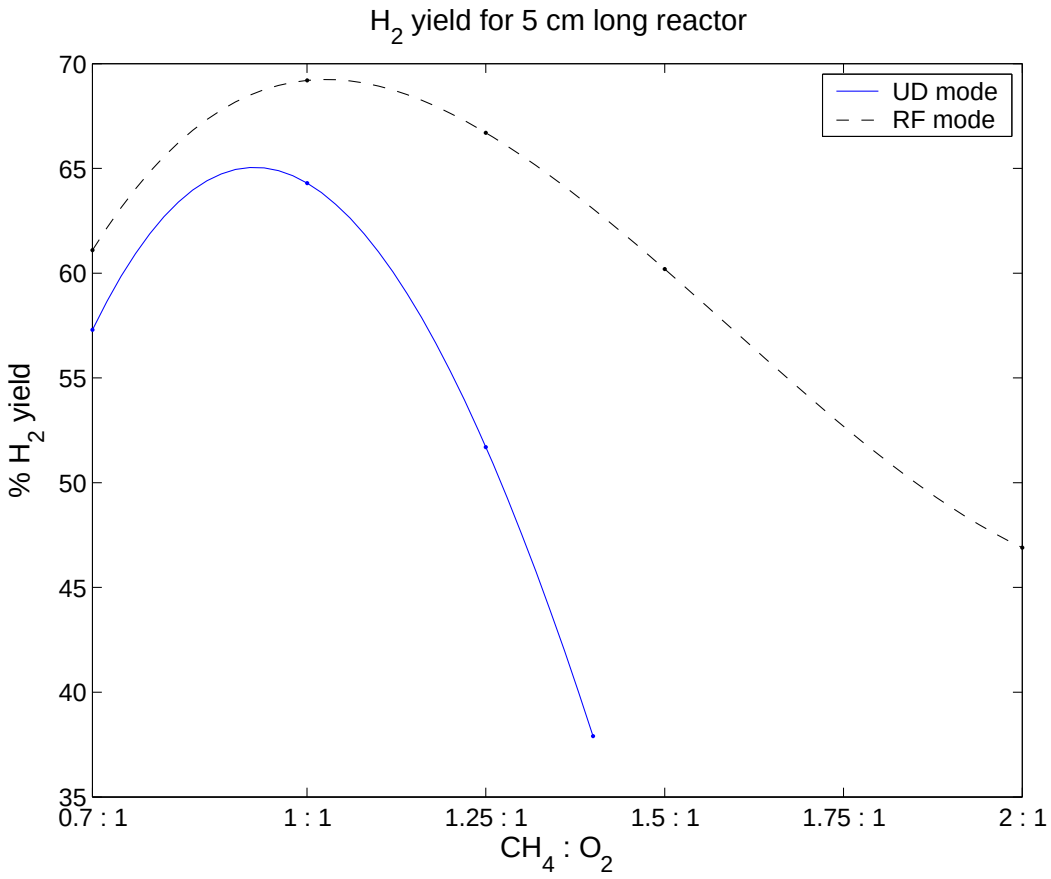

Figure 12: Effect of varying feed ratios in a $5cm$ reactor. RF operation (– –) provides significant improvement over UD (—) operation in this shorter reactor.

3.3 Analysis of the Reverse-Flow Operation

In this section, we attempt to provide plausible explanations of process intensification in a microreactor under RF operation over UD operation. As the reaction and diffusion time scales are fast, the reactor operation is governed by the thermal dynamics of the microreactor. The time scale of reaction heat release ($\tau_{Hrxn} = 3.2\,sec$) and the time scale of thermal relaxation of a volume element of the reactor ($\tau_{th} = 168\,sec$) are critical parameters that provide approximate handles for selection of switching time $\tau_{c/2}$. There are three different possibilities for selecting the switching time:

- switching faster than time of heat release by reaction ($\tau_{c/2} \leq \tau_{Hrxn}$),
- switching slower than the thermal relaxation time scale ($\tau_{c/2} \geq \tau_{th}$), and

Table 7: Reactor performance with various inlet feed ratios for poorly insulated reverse-flow reactor case with switching time of 5 *sec*.

Case	$CH_4 : O_2$ ratio	$T_{s,max}$	$T_{s,out}$	$\%H_2$ yield
1	0.7 : 1	1598	750	47.2
2	1 : 1	1566	696	44.5
3	1.25 : 1	1474	634	38.0
4	1.5 : 1	1445	610	31.6
5	2 : 1	1436	595	25.2

- switching between the two time scales ($\tau_{Hrxn} < \tau_{c/2} < \tau_{th}$).

Investigating the reactor conditions for various times allows us to analyze the reactor operation.

3.3.1 Nominal Reactor: Favorable thermodynamic conditions

As the first example, we investigate the performance of the microreactor for the base case (case 2 in Table 6). As the approximate time scales for the reactions are smaller than the advection time scale, the reactions attain equilibrium within the reactor. The rates of oxidation, reforming and water gas shift at UD steady state are plotted in Fig. 13 with the temperature profile shown in the background as a thick gray line. Most of the reactions take place near the temperature maxima, which occurs about 3.5 *cm* from the reactor inlet, while the rest of the reactor is utilized in reaching equilibrium. The portion of the reactor upstream of this zone gets cooled by the incoming stream and virtually goes unutilized as no reaction takes place there.

Figure 14 shows the temperature profile at various times after the flow reversal for a half-cycle time of $\tau_{c/2} = 200$ *sec*. Immediately after the flow reversal, the temperature at the reactor inlet (outlet in the previous half-cycle) is high, while that at the outlet is low. As the time progresses, the temperature at the inlet drops due to the incoming cold feed and that at the outlet increases due to the heat released on reaction and heat transfer from the hot gas flow. The time taken for this is governed primarily by the time scale of solid

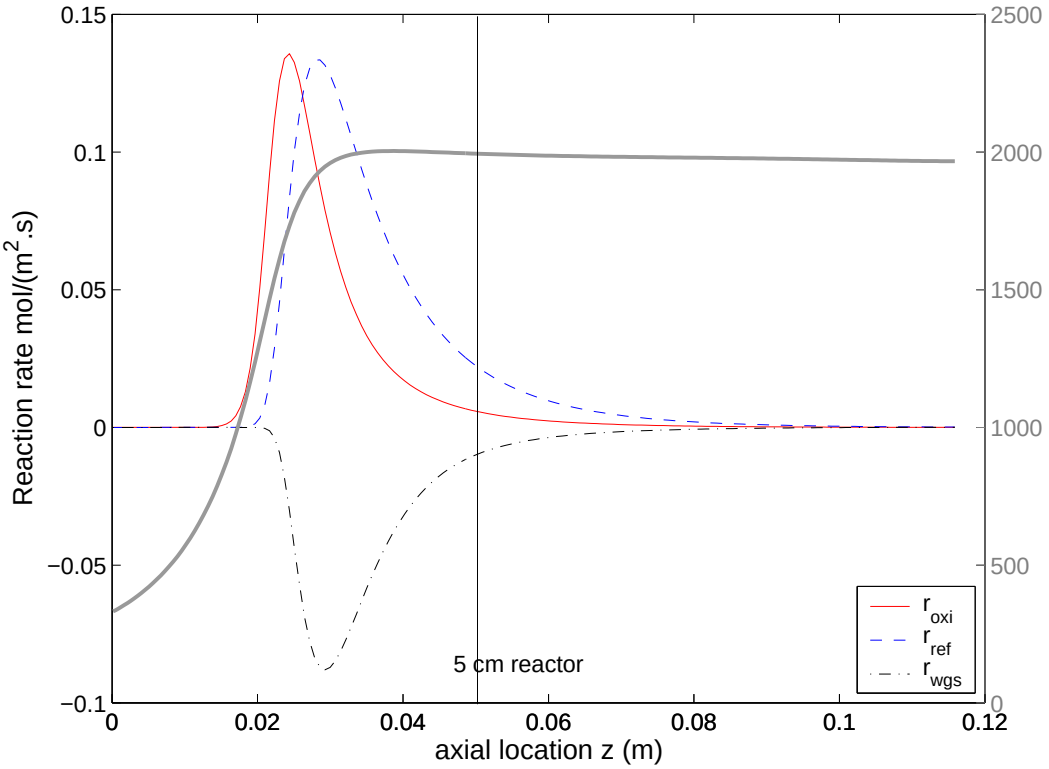

Figure 13: Reaction rates at UD steady state for the Base Case. The temperature profile is shown as thick gray line. Most of the reactions take place approximately 3.5 cm from the reactor inlet. Profiles for the shorter 5 cm-reactor are similar. The vertical line represents the 5 cm mark.

thermal inertia (τ_{th}) as well as the time scale of reaction heat release (τ_{Hrxn}). At 5 sec after the flow reversal (dotted line), which is an order of magnitude lower than τ_{th}, the outlet temperature is still quite low. As the time progresses, the inlet temperature drops and the outlet temperature increases progressively (dashed line – 50 sec after flow reversal). At 200 sec, which is greater than τ_{th}, the temperature profile is similar qualitatively to the UD steady state. Thus, if the switching time is large, *e.g.* $\tau_{c/2} = 200\,sec$ comparable to τ_{th}, the reactor has enough time to respond thermally to the temperature changes. As a result, the conditions in RF operation approach that in UD operation, thus hardly any change in hydrogen yield is observed at higher switching time. This is illustrated in Fig. 15, where the RF yield asymptotically approaches the UD yield as $\tau_{c/2}$ is further increased beyond τ_{th}.

The temperature profile (thick grey line) and the rates of reactions in RF operation

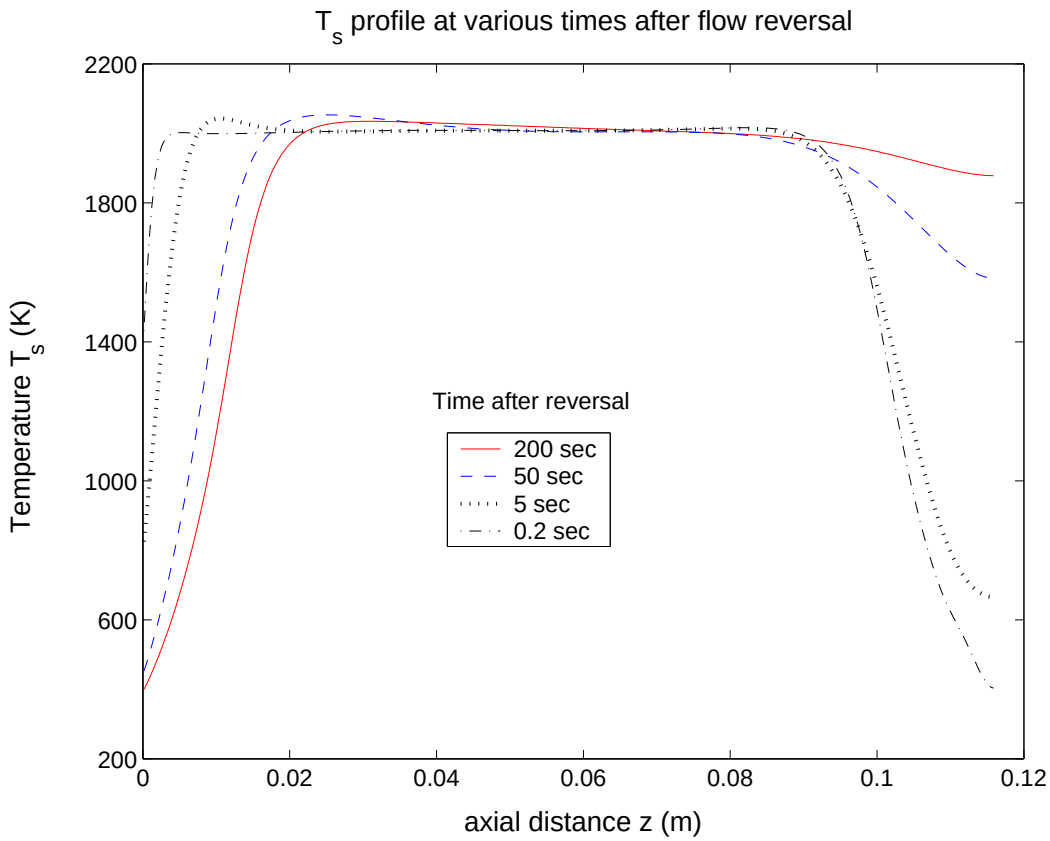

Figure 14: Temperature profiles at various times after flow reversal in the reactor with switching time of $\tau_{c/2} = 200\,sec$.

with a fast switching time of $5\,sec$ are shown in Fig. 16. A low temperature region that appears at the reactor end is thermodynamically favorable for the water gas shift (WGS) reaction, resulting in an improved H_2 yield. Although low temperature is unfavorable for the endothermic reforming reaction, the equilibrium constant for reforming reaction is still high enough to ensure that the reaction essentially proceeds to completion and does not adversely affect the H_2 yield. As the switching time increases, there is a decrease in H_2 yield, with the optimal switching time in the range $2 - 5\,sec$. Switching at very high frequencies ($< 1\,sec$) is also undesirable as hydrogen concentration requires some time to respond and attain pseudo-steady state after the port switching.

To ensure that indeed the favorable thermodynamic conditions for WGS reaction were responsible for the observed phenomenon, we simulated the reactor assuming no WGS

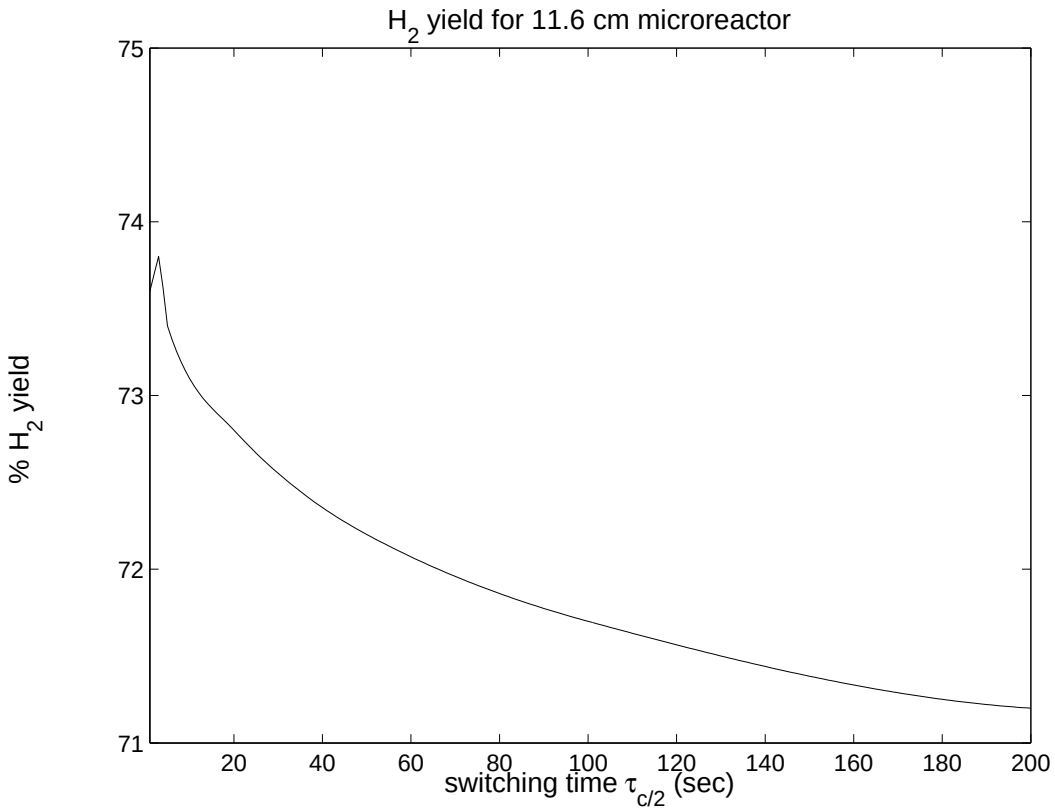

Figure 15: Hydrogen yield in RF operation as a function of the switching time. Fast switching of the reactor provides greater hydrogen yields.

occurs within the reactor. With WGS 'turned off', the H_2 yield on RF operation was similar to that of UD operation (a small decrease in H_2 yield was observed on the account of reforming reaction).

Another interesting outcome is that the "NK" kinetics model considered in part-1 of this report [56] could not reproduce the observed differences in UD and RF operations. Figure 17 shows the rates of WGS: thick lines represent RF operation, thin lines represent UD operation; solid lines represent the GOS model and dashed lines represent the NK model. The rate of WGS reaction in the NK model is an order of magnitude lower than the GOS model. In RF operation, no significant WGS reaction occurs in the small end region of lower temperature in the reactor (thick, dashed line). As a result, the difference in UD and RF yields is not significant for the NK model.

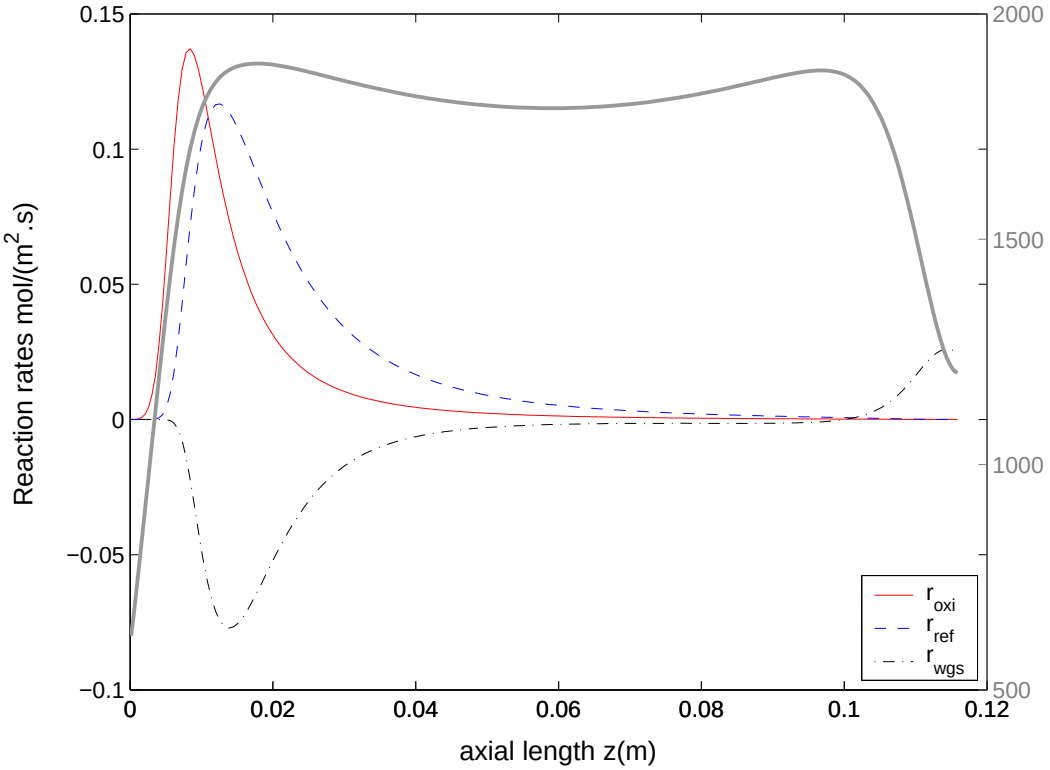

Figure 16: Rates of reactions and temperature profile (thick gray line) just prior to input-output port switching for a switching time of 5 *sec*. High temperature central region allows reactions to reach equilibrium. Lower temperature at the reactor end favors water gas shift reaction.

3.3.2 Better thermal utilization in a short reactor

Figure 18 shows the hydrogen yield for RF operation in a 5 *cm*-long reactor as a function of the time of flow reversal. As in the previous case, faster switching in the reactor gives higher hydrogen yield, while the hydrogen yield asymptotically approaches that in the UD operation at larger switching times. A clear maxima is observed at switching time of about 4 *sec*.

For the shorter reactor, the advection time scale is comparable to that of the reforming reaction. The rates of reactions and temperature profile are similar to those seen in the previous example (Fig. 13); the temperature maxima is observed at 3.5 *cm* from the reactor inlet. Consequently, the reactor length is not sufficient for the reforming and WGS reactions to reach equilibrium. This results in a lower hydrogen yield in UD operation. As

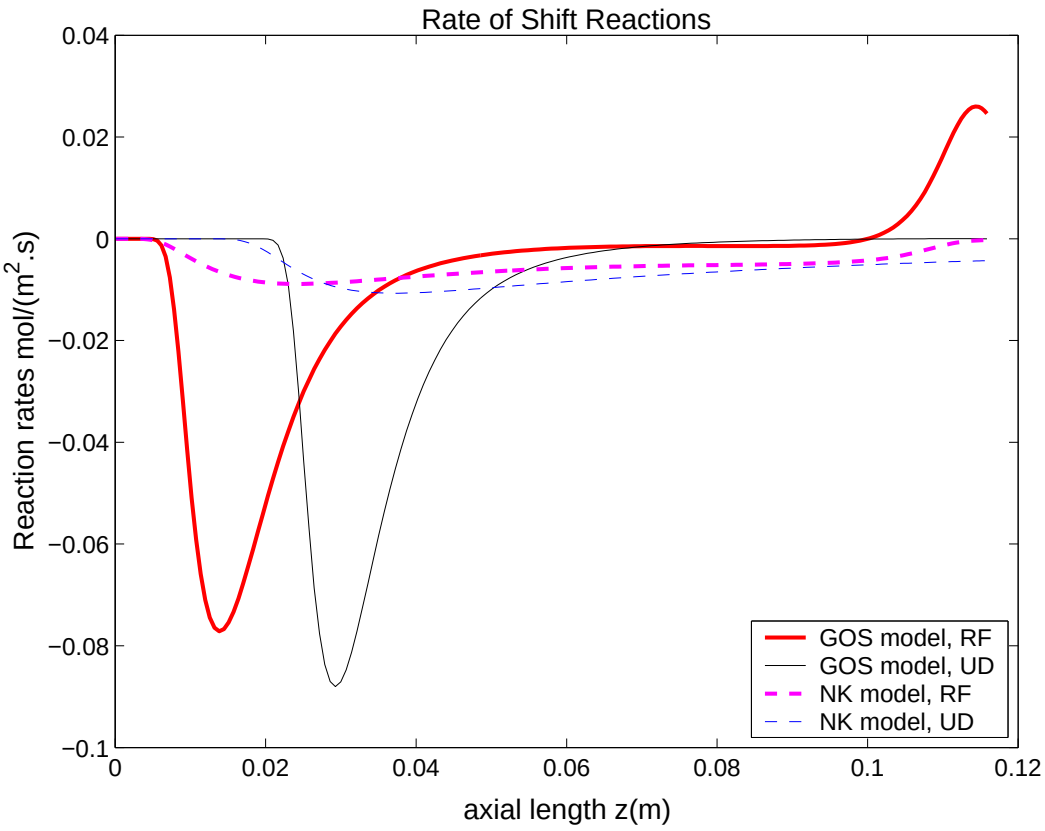

Figure 17: Rate of water gas shift reaction in UD (thin lines) and RF (thick lines) operation. For the GOS model (—), low T_s at the reactor end favors shift reaction, while it is insignificant for NK model (– –).

temperature peak occurs close to the reactor exit, the entire length of the microreactor is not properly utilized.

Reverse-flow operation with fast switching results in a more uniform temperature profile at the center of the reactor. Figure 19 shows the temperature profiles within the reactor just prior to switching of the input and output ports for three different switching times of $1\,sec$ ($\tau_{c/2} < \tau_{Hrxn}$), $3\,sec$ ($\tau_{c/2} \approx \tau_{Hrxn}$) and $20\,sec$ ($\tau_{c/2} > \tau_{Hrxn}$). The temperature profile for fast switching time of $\tau_{c/2} = 1\,sec$ is shown as a solid line in the figure. As the switching time is increased, the peak temperature increases, resulting in an increase in conversion to hydrogen. This happens approximately until switching time reaches the time scale of reaction heat release. When the switching time is further increased, the reactor enters a wave development stage [25], wherein a single reactor hot-spot is formed and it

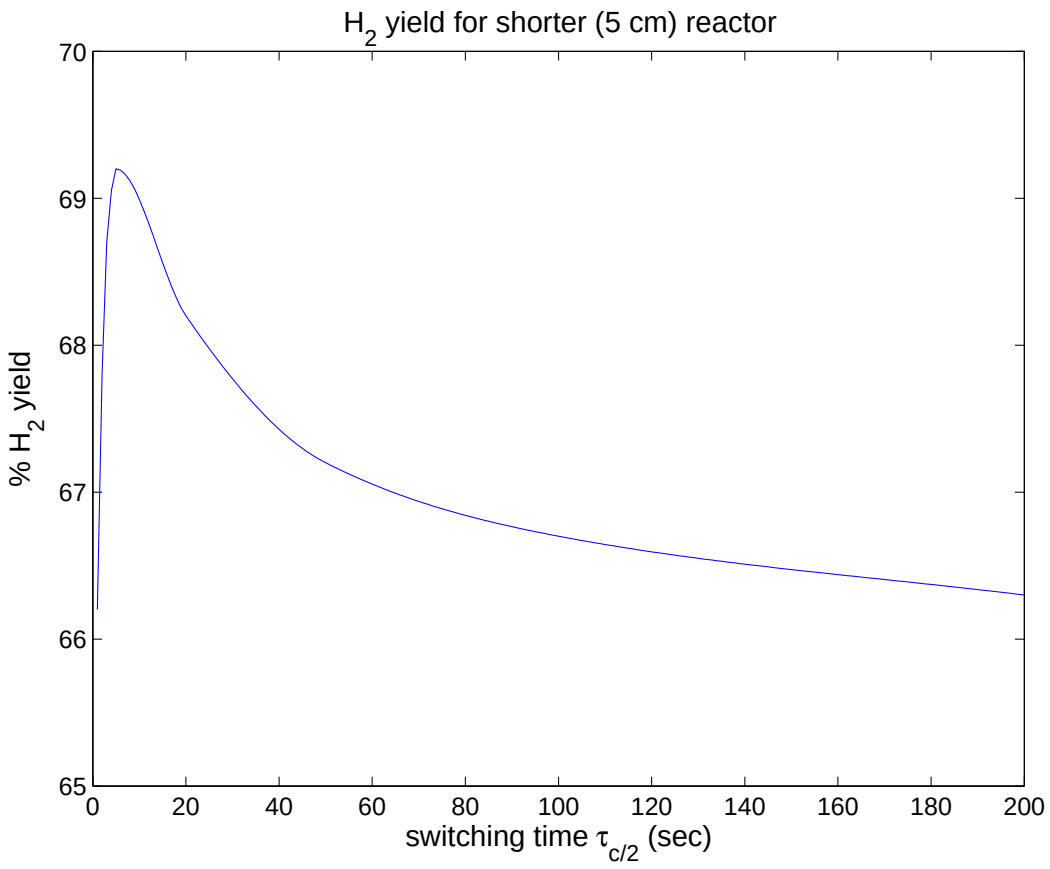

Figure 18: Hydrogen yield in RF operation as a function of the switching time for a $5\,cm$ reactor. The maxima observed at $\tau_{c/2} \approx 4\,sec$ corresponds to the time scale of reaction heat release.

starts migrating towards the center of the reactor. In this stage, the output hydrogen concentration starts falling as the reforming reaction is unable to proceed to completion as the hot-spot approaches its UD steady-state location. Consequently, a maxima in the hydrogen yield is observed at $\tau_{c/2} = 4\,sec$, which corresponds to the approximate time-scale of reaction heat release τ_{Hrxn}, due to the optimal thermal utilization of the reactor.

3.3.3 Using reactor as a regenerative heat exchanger

One of the most widely studied application of RF operation is to utilize the reactor structure as a regenerative heat exchanger to trap a creeping thermal wave within the reactor and maintain autothermal operation in systems where the heat released on reactions is insufficient to maintain autothermal UD operation. This situation is indeed observed in

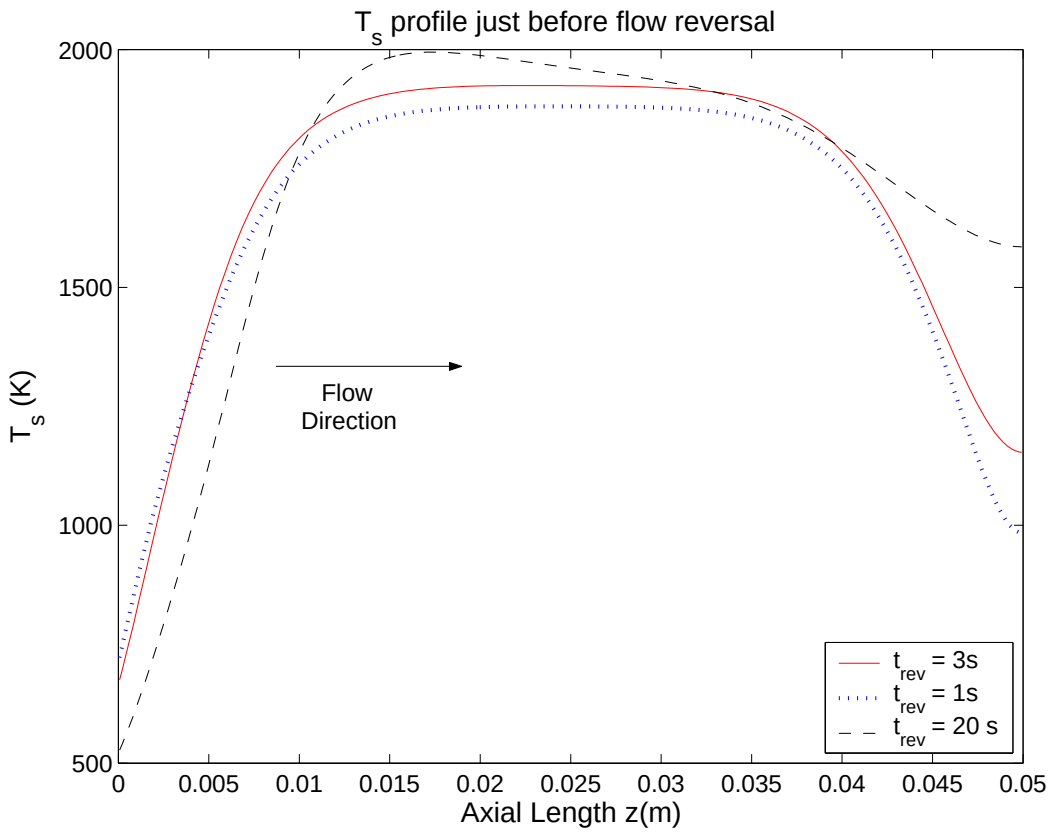

Figure 19: Temperature profile for various switching times in the $5\,cm$ reactor after attaining periodic steady state.

simulations if the microreactor is poorly insulated and has significant heat losses to the surroundings. Methane oxidation is unable to provide adequate heat to maintain autothermal operation in presence of heat losses. If the reactor temperature exceeds the ignition temperature, oxidation reaction takes place and a hot spot is formed. The incoming cold stream cools the reactor upstream of this hot spot and pushes it further inside the reactor. This results in a thermal wave that travels in the direction of flow. The flow direction is switched before the wave exits the reactor, trapping it within the reactor to provide autothermal operation.

Fast switching in this case results in significantly lower peak temperatures, due to the large heat losses. Consequently, the reactions do not proceed to completion and the hydrogen yield is low. As the switching time is increased, the peak temperature increases, a thermal wave develops and starts propagating along the length of the reactor. Switching

at the wave propagation stage gives the maximum yield. Hydrogen yield is plotted as a function of switching time $\tau_{c/2}$ in Fig. 20. Optimum RF performance is obtained as the switching time approaches the time scale of thermal inertia. Increasing the switching time beyond τ_{th} does not cause any significant increase in hydrogen yield. The maximum limit for switching time is given by the time required for the reaction zone to travel through the reactor, which was found to be approximately 1200 *sec* for this case.

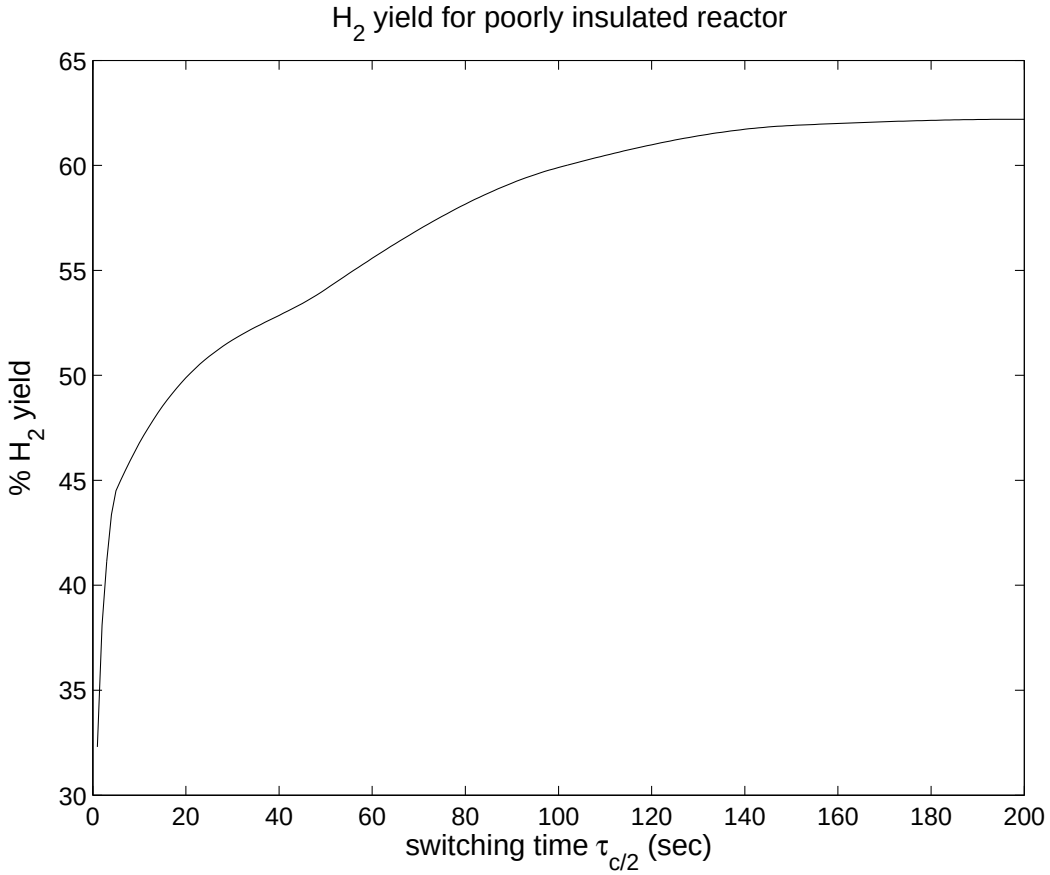

Figure 20: Hydrogen yield in RF operation as a function of the switching time for poorly insulated reactor case. Infrequent switching on the scale approaching the time scale for the reactor thermal inertia provides maximum yield for this case.

3.4 Conclusions

Parametric simulation results of autothermal partial oxidation of methane in a microreactor were presented in this paper. It was demonstrated that reverse-flow operation of the reactor

provides better hydrogen yields and lower temperatures for most cases considered. An analysis of the observed improvement in the reactor performance in reverse-flow operation over unidirectional operation was presented and approximate guidelines on operating the reverse-flow reactor based on simple time scale analysis were developed. Specific results of this work include:

- The thermal dynamics of the reactor was found to be the dominant effect in this system, which can be exploited through reverse-flow operation.
- The time scale of heat release due to reactions was found to be an important parameter for the system. The optimum reactor yield was obtained when the switching time was close to this time scale.
- In addition to some improvements in reactor yields, the main advantages of the reverse-flow operation were lower reactor temperature and robustness of the reactor performance under various operating conditions. The unidirectional autothermal steady state disappeared at higher inlet velocities, higher methane content in the feed and poor reactor insulation (significant heat losses). However, autothermal operation was maintained in the reverse-flow mode.
- The improvement in hydrogen yield in RF operation was much greater in case of the shorter $5\,cm$-long reactor.
- Three different cases for process intensification in reverse-flow operation were observed in the system.
  - For the $11.6\,cm$-long reactor, the improvement in hydrogen yield occurred due to better thermodynamic conditions at the reactor exit. In this case, fast switching ($\tau_{c/2} = 2$ to $5\,sec$) was found to be optimal.
  - Better thermal utilization of the shorter reactor was responsible for improved hydrogen yield. The optimal reactor performance was obtained when the switching time approached the time scale of reaction heat release.

  - For the large heat loss case, the microreactor was used as a regenerative heat exchanger to trap the travelling thermal wave by flow reversal.

- Not all cases are guaranteed to provide improvement in hydrogen yield on reverse-flow operation. For $CH_4 : O_2 = 1.25 : 1$ and higher, the unidirectional steady state was in fact better than the reverse-flow operation.

CHAPTER IV

SENSITIVITY AND OPERABILITY ANALYSIS OF THE MICROREACTOR

This chapter considers operability analysis and design of a microreactor for hydrogen generation via methane partial oxidation. The microreactor is used in both unidirectional (UD) and reverse-flow (RF) modes of operation. We show that the reverse-flow operation is more robust to various changes in reaction and transport parameters. Autothermal operation can be maintained only in the RF operation and not in the UD operation at higher hydrogen throughput or with partial deactivation of the catalyst. Increasing the catalyst loading and using reactor with higher solid thermal conductivity is shown to increase the region of autothermal UD operation. An analysis of model sensitivity to the kinetic parameters is performed. Based on this, a rational scheme for improved catalyst placement in the RF reactor is proposed. Finally, an opposed-flow operation is suggested to improve reactor robustness. It's performance is compared with that of the reverse-flow operation.

4.1 Introduction

Forced unsteady state operation (FUSO) has been shown to provide improved performance over the conventional steady state operation in many chemical systems [74, 75]. An example of FUSO is the reverse-flow (RF) operation of tubular or fixed bed reactors [14, 25, 43]. The RF operation is achieved by periodically switching the input and output ports to reverse the flow direction. Recently, the advantages of the reverse flow (RF) operation over the unidirectional (UD) operation for methane partial oxidation in a microreactor were demonstrated experimentally [59]. A theoretical study aimed at explaining the physical origins of the observed improvement was presented in the previous chapters. Under various different conditions, the RF operation provided improved performance due to several reasons, such as favorable thermodynamic conditions, better thermal utilization of the reactor and using

the reactor as a regenerative heat exchanger [57].

In Chapters 2 and 3, we observed that the RF operation does not necessarily provide higher hydrogen yields than the UD operation under all circumstances. For example, under nominal operating conditions, the UD operation provided better hydrogen yield than the RF operation for $CH_4 : O_2$ ratio of $1.2 : 1$ and higher. However, at higher velocities and higher amounts of methane in the feed, a travelling thermal wave was formed and the autothermal UD operation was no longer possible. On the other hand, the autothermal operation could be maintained in the RF mode. This condition (disappearance of an autothermal UD state) is likely to further deteriorate with several hours of reactor operation due to a decrease in the effectiveness of the catalyst (catalyst deactivation). The aim of this paper is to provide an operability analysis for the UD and RF operations of the microreactor, with respect to both hydrogen yield and hydrogen throughput within the reactor. Design changes to improve the robustness of the UD operation will also be considered, so that higher amounts of hydrogen is generated under increased load conditions.

Eigenberger and Nieken [31] proposed various reactor designs, which integrate regenerative heat exchange with catalytic oxidation, to maintain autothermal operation for catalytic air purification of very dilute fuel effluent streams. In the case of the RF operation, the thermal energy liberated by the mildly exothermic process gets stored in the solid reactor body due to its high heat capacity and large time scale of thermal relaxation. This heat is released to the inlet cold gas after the flow reversal. Thus, the reactor itself acts as a regenerative heat exchanger and an autothermal operation is maintained. In our microreactor system, we found that the overall thermal dynamics of the system were critical in obtaining performance improvement [56]. Motivated by the similarity between Eigenberger and Nieken's system and a RF reactor, we consider an alternate design of reactor, which we term as "opposed flow" (OF) system. The key idea of an OF reactor is similar to that of a countercurrent heat exchanger, as shown in Figure 21. The reactant gases flow in opposite directions in adjacent channels. If we assume that there is no thermal interaction between the adjacent channels, the temperature profiles will be as shown in the top plot in Figure 21. The temperature profile in an actual OF reactor can be thought of some superposition of

the two UD profiles. As shown in the bottom plot, the temperature profile is similar to that observed in the RF operation. Several authors have previously shown that for infinitely large coefficient of heat transfer, internal loop reactors and countercurrent opposed flow reactors are analogous to the reverse flow reactor with infinitely fast switching [60, 61]. In a recent special issue on FUSO [94], Sheintuch and Nekhamkina [96] presented simulation and experimental results comparing these systems. This motivates us to consider the OF operation as an alternative to the RF operation to maintain robust autothermal reactor operation.

The rest of this report is organized as follows. Section 4.2 considers parametric sensitivity analysis of the reactor to identify the key transport and reaction parameters affecting the system, and their effect on the optimal feed conditions and switching time. A rational scheme for optimal placement of various catalysts, which exploits the spatial temperature patterns in the RF reactor, is presented in section 4.3. Section 4.4 considers operability analysis of the UD and RF operations; the response to hydrogen throughput changes, the robustness of the reactor to catalyst deactivation, the effect of catalyst loading and thermal conductivity of the reactor are considered. Section 4.5 proposes some design changes in the system for improved UD operation. Finally, results of this work are summarized in the concluding section.

4.2 Sensitivity Analysis

The aim of the sensitivity analysis is to identify the various model parameters that have a significant influence on the reactor operation. The reactor model was discussed in our previous paper [56]. The key system parameters are the kinetic rate constants, the transport coefficients, and the parameter α that splits heat of reaction between gas and solid energy conservation equations. As the system is limited by the rate of catalytic reactions, the mass transfer coefficient k_g does not have a strong influence on the reactor performance. The heat transfer coefficient is also very high due to the small channel diameters. Hence, the temperatures of the solid and gas (T_s and T_g) do not differ significantly. As a result, the parameter α was also found to have little effect on the reactor performance. Also, the effect

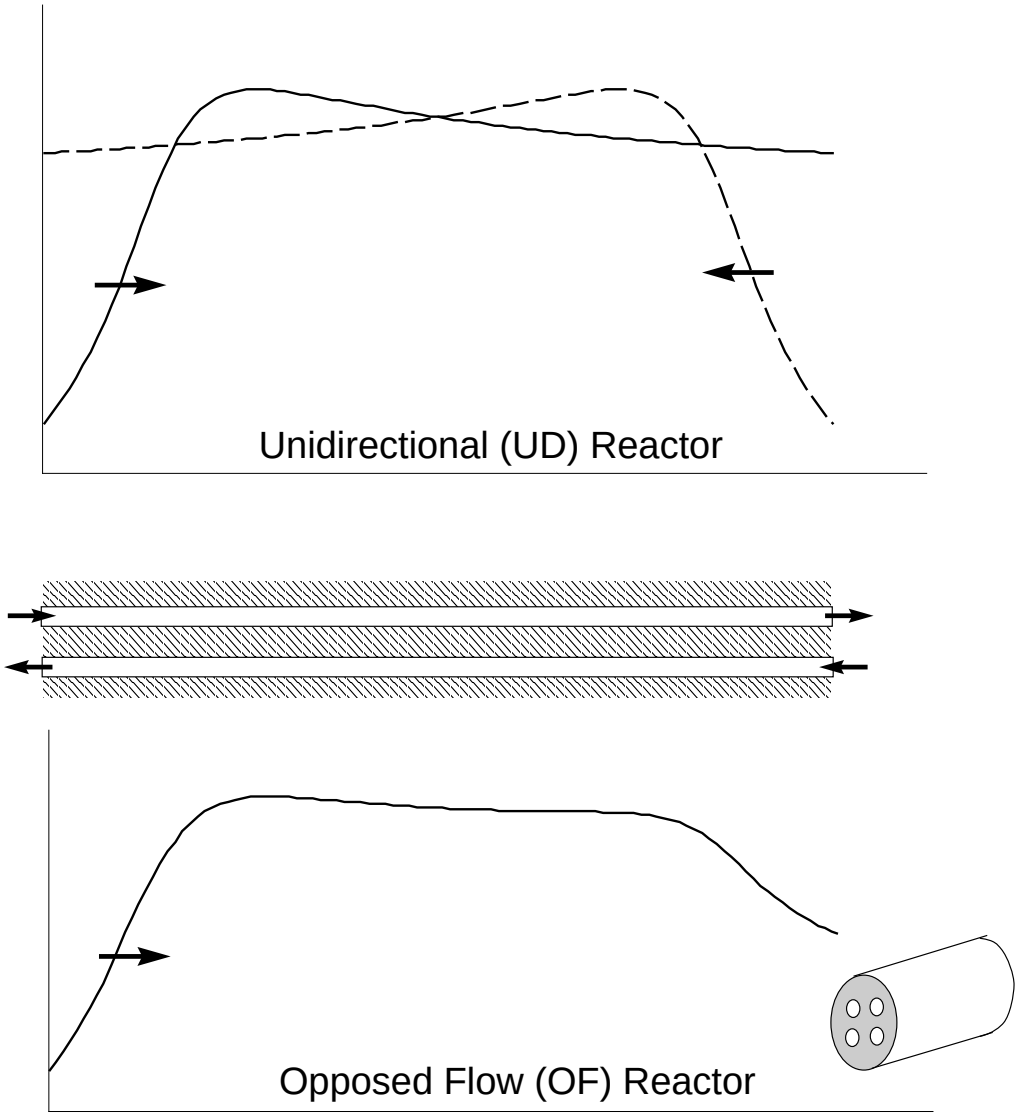

Figure 21: A cartoon explaining the concept of an opposed flow (OF) reactor.

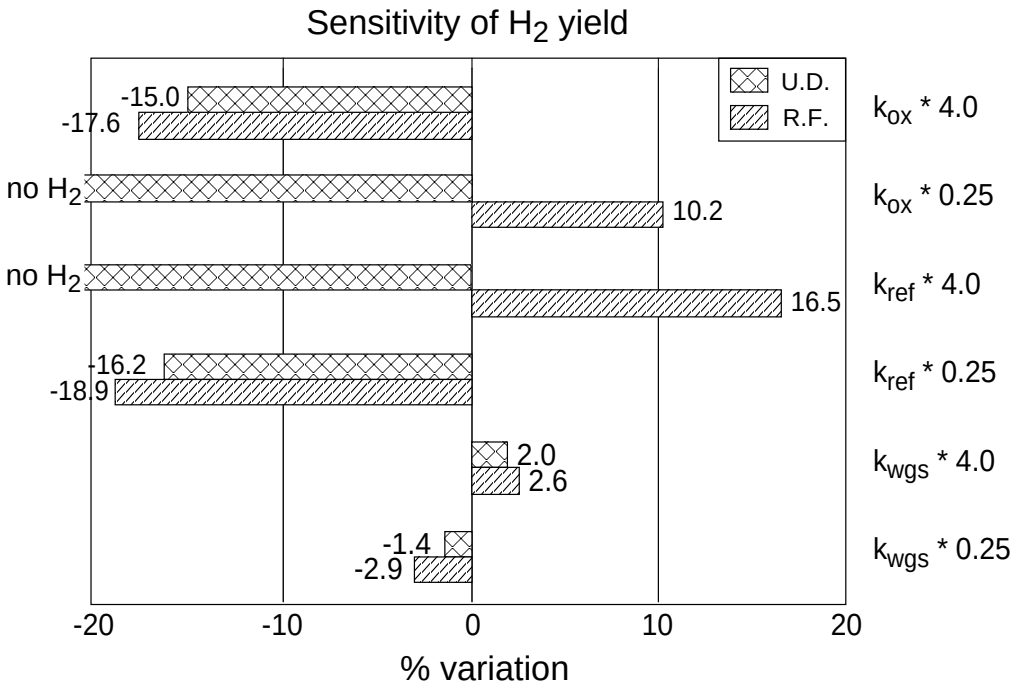

Figure 22: Sensitivity of hydrogen yield to variations in kinetic constants

of varying the heat transfer coefficient h_v, although higher than those of k_g and α, was still insignificant. Therefore, we only consider variations in kinetic parameters and its effect on reactor performance.

4.2.1 Sensitivity to kinetic parameters

The kinetic constants of oxidation, reforming and water gas shift reactions were varied individually by a factor of 4. The sensitivity of hydrogen yield on the variations in these parameters for the UD and RF operations is shown in Figure 22. The autothermal UD state disappeared when the rate of oxidation was reduced or the rate of reforming reactions was increased. On the other hand, increasing the rate of oxidation or decreasing the rate of reforming reaction stabilized the UD operation, although hydrogen yield was significantly lower. Variations in the rate of water gas shift reaction did not result in significant variations in hydrogen yield.

Additionally, we also studied how the variations in the kinetic constants affected the

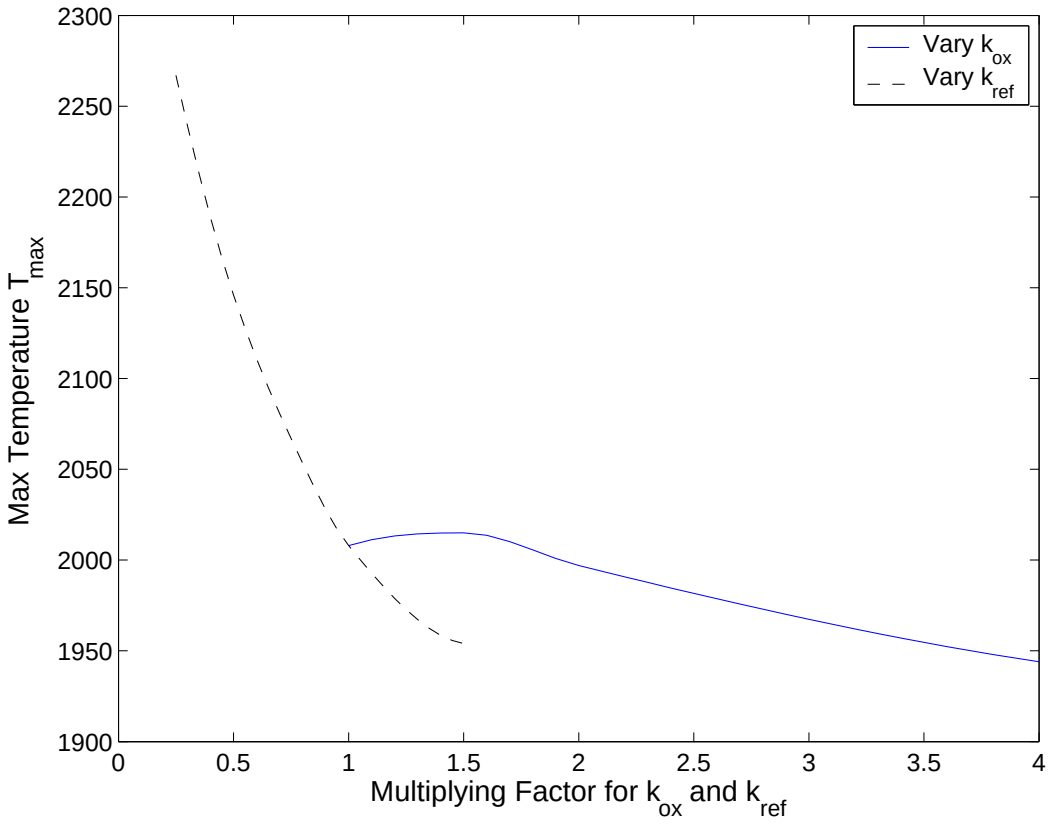

Figure 23: Effect of varying kinetic constants for oxidation and reforming reactions. The maximum temperature increases monotonically as k_{ref} is decreased. When k_{ox} is increased, the temperature increases initially. However, beyond a certain point, the maximum temperature decreases again.

temperature profiles in the reactors. An increase in the rate of the exothermic oxidation and a decrease in the rate of endothermic reforming reaction are expected to increase the maximum temperature in the reactor. Figure 23 shows that the maximum temperature increases, as expected, when the reforming rate is progressively decreased. In contrast, when the oxidation rate is increased, the maximum temperature increases initially and the temperature peak shifts towards the entrance of the reactor. A further increase in the temperature pushes the temperature peak further towards the reactor entrance. As the solid temperature at the reactor entrance increases, so does the loss of energy by convection and radiation at the reactor entrance. Thus, beyond a certain point, the temperature starts decreasing as the rate of oxidation is increased. Figure 24 shows the temperature profiles for various kinetic constants.

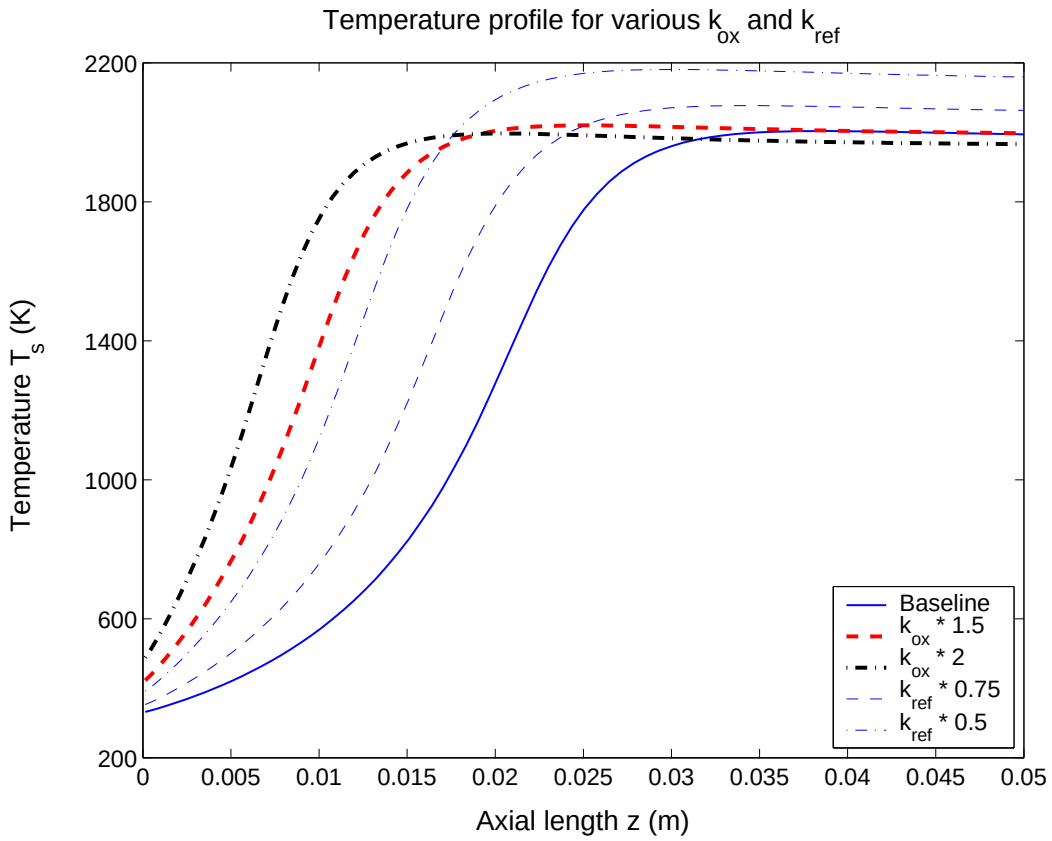

Figure 24: Steady state temperature profiles for various values of kinetic rate constants. Thick lines: increased k_{ox}; thin lines: reduced k_{ref}. At higher k_{ox}, the temperature profiles shift towards the entrance, resulting in a lower $T_{s,max}$.

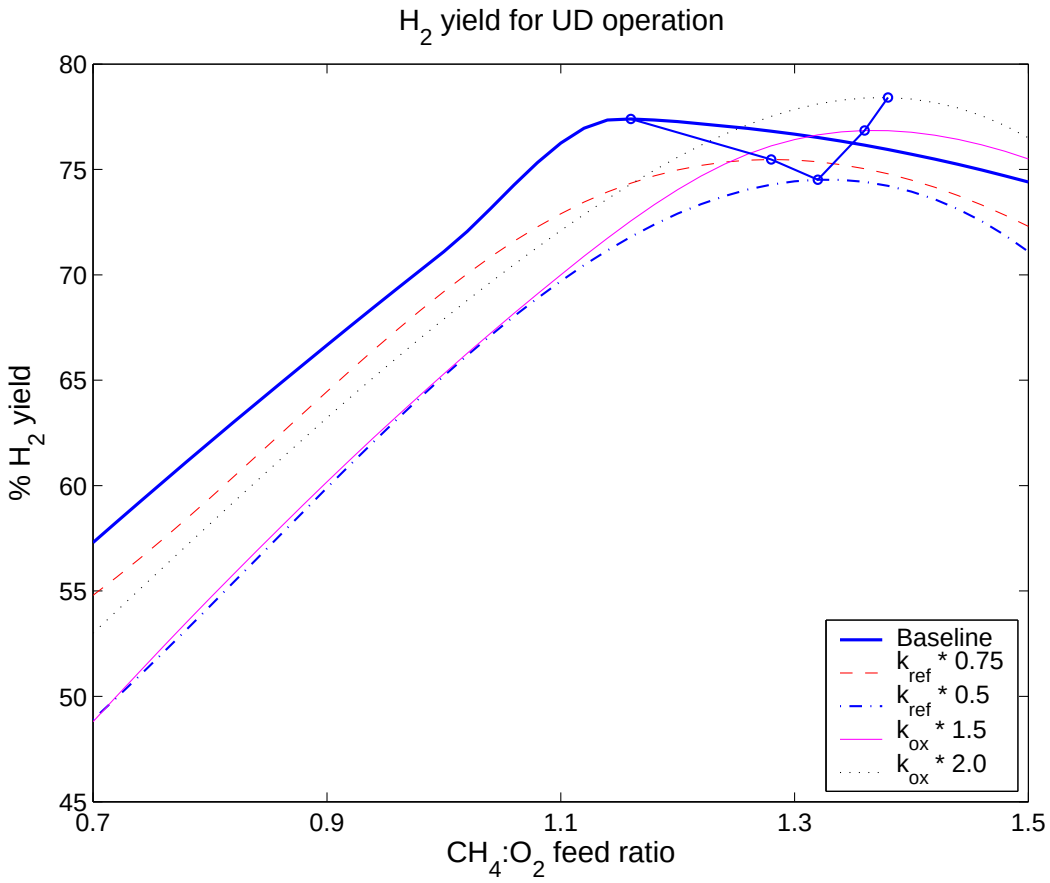

Figure 25: Hydrogen yield for UD operation as a function of inlet feed ratio for various values of kinetic constants k_{ox} and k_{ref}.

Based on the observations presented here, further analysis of reactor operability is focussed primarily on the variations in kinetic rate constants of oxidation and reforming reactions.

4.2.2 Optimal feed conditions

The variations in the optimal $CH_4 : O_2$ feed ratio with the kinetic parameters for both the UD and RF operations are analyzed in this section. The rate constants of oxidation and reforming reactions were increased and decreased individually up to a factor of 2 and the simulations were performed for various choices of inlet feed ratios. As noted earlier, we found that decreasing the rate of oxidation k_{ox} or increasing the rate of reforming reaction k_{ref} resulted in disappearance of the autothermal UD steady state. Figure 25 shows the

Table 8: Variations in reactor performance with changes in kinetic parameters

	Reverse Flow			Unidirectional		
	$[\%H_2]_{1:1}$	$[\%H_2]_{max}$	Opt Feed	$[\%H_2]_{1:1}$	$[\%H_2]_{max}$	Opt Feed
Base	73.5	77.5	1.16 : 1	71.1	77.5	1.20 : 1
$k_{ox} * 2.0$	66.6	70.5	1.16 : 1	65.3	78.4	1.36 : 1
$k_{ox} * 1.5$	69.5	73.5	1.18 : 1	67.1	76.8	1.38 : 1
$k_{ox} * 0.75$	76.9	79.1	1.16 : 1	tw	tw	—
$k_{ox} * 0.5$	78.9	80.0	1.16 : 1	tw	tw	—
$k_{ref} * 2.0$	80.5	82.3	1.16 : 1	tw	tw	—
$k_{ref} * 1.5$	77.8	80.4	1.18 : 1	tw	tw	0.7 : 1
$k_{ref} * 0.75$	71.0	74.7	1.20 : 1	69.2	75.5	1.28 : 1
$k_{ref} * 0.5$	67.0	71.0	1.18 : 1	65.2	73.5	1.32 : 1

$[\%H_2]_{1:1}$: % hydrogen yield for the feed ratio $CH_4 : O_2 = 1 : 1$

$[\%H_2]_{max}$: Maximum % hydrogen yield obtained

Opt Feed : Optimal $CH_4 : O_2$ feed ratio

variations in hydrogen yields as a function of $CH_4 : O_2$ feed ratios in the UD operation. The maximum hydrogen yields are also indicated for the various kinetic constants. The optimal feed ratio varied between 1.1 : 1 to 1.4 : 1. An increase in k_{ox} and a decrease in k_{ref} resulted in an increase in the relative amount of CH_4 in feed. These results are summarized numerically in Table 8. The table shows hydrogen yields for nominal value of $CH_4 : O_2 = 1 : 1$, the optimal hydrogen yields and the optimal feed ratios for the various changes in the kinetic parameters discussed earlier.

In contrast to UD, autothermal operation was maintained in the RF mode. Figures 26 and 27 show the variations in hydrogen yields for different values of k_{ox} and k_{ref} respectively, for the RF operation. Autothermal operation was possible in the RF mode for all the variations in the kinetic parameters. Unlike the UD operation, the optimal feed ratio $CH_4 : O_2$ in the RF operation stayed in the vicinity of 1.16 : 1 for various changes in the kinetic constants. Thus, we see that the optimal RF operation does not change with changes in the system parameters. These results are summarized numerically in Table 8.

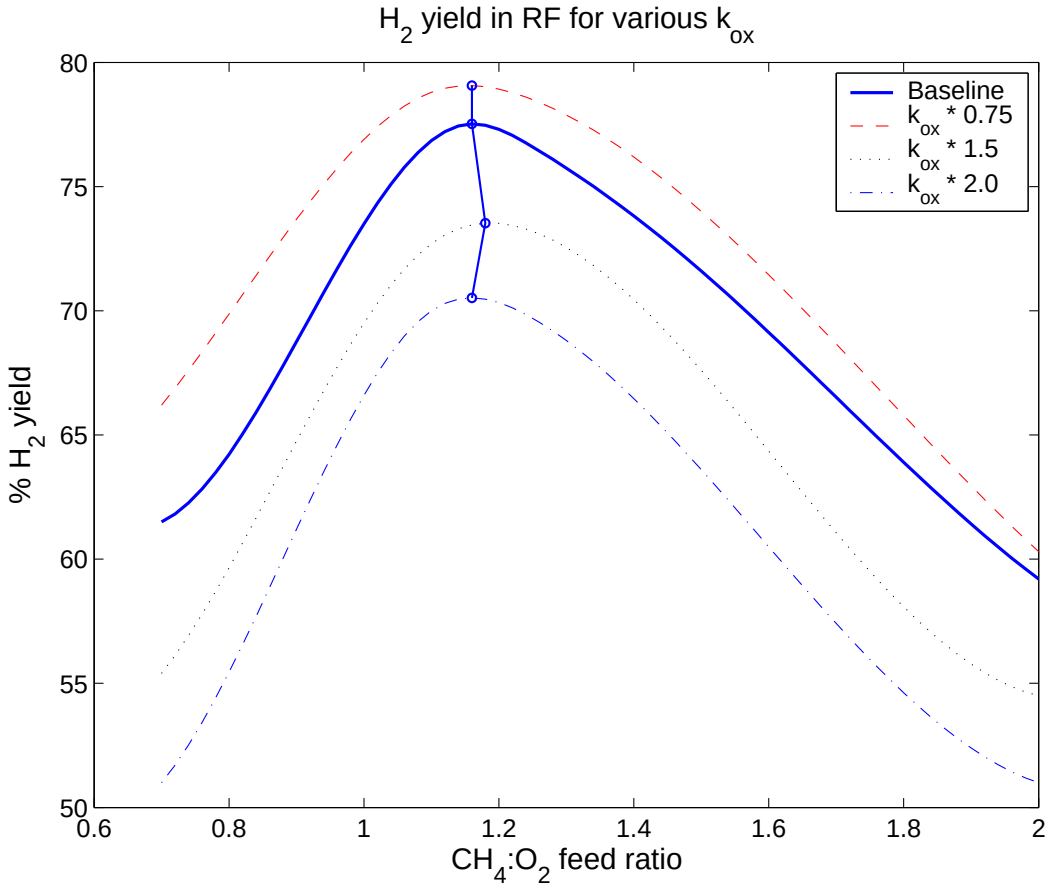

Figure 26: Hydrogen yield in the RF reactor as a function of inlet feed ratio for various values of kinetic constant k_{ox} for the oxidation reaction.

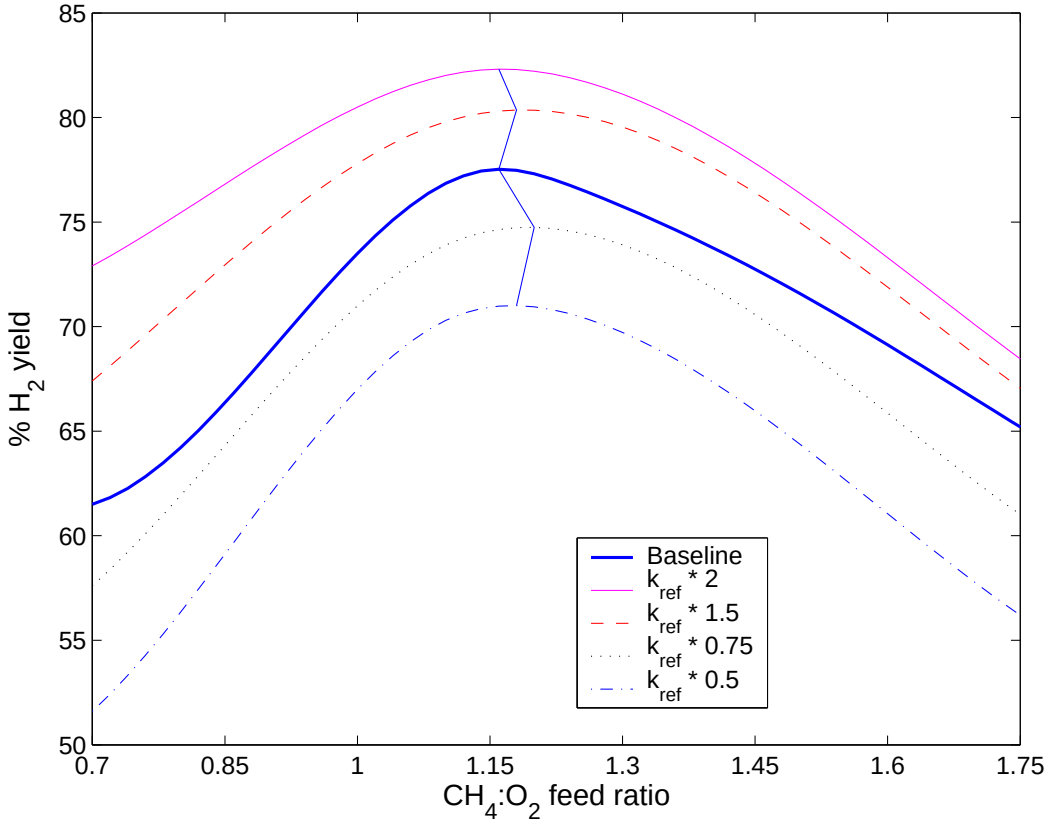

Figure 27: Hydrogen yield in the RF reactor as a function of inlet feed ratio for various values of kinetic constant k_{ref} for the reforming reaction.

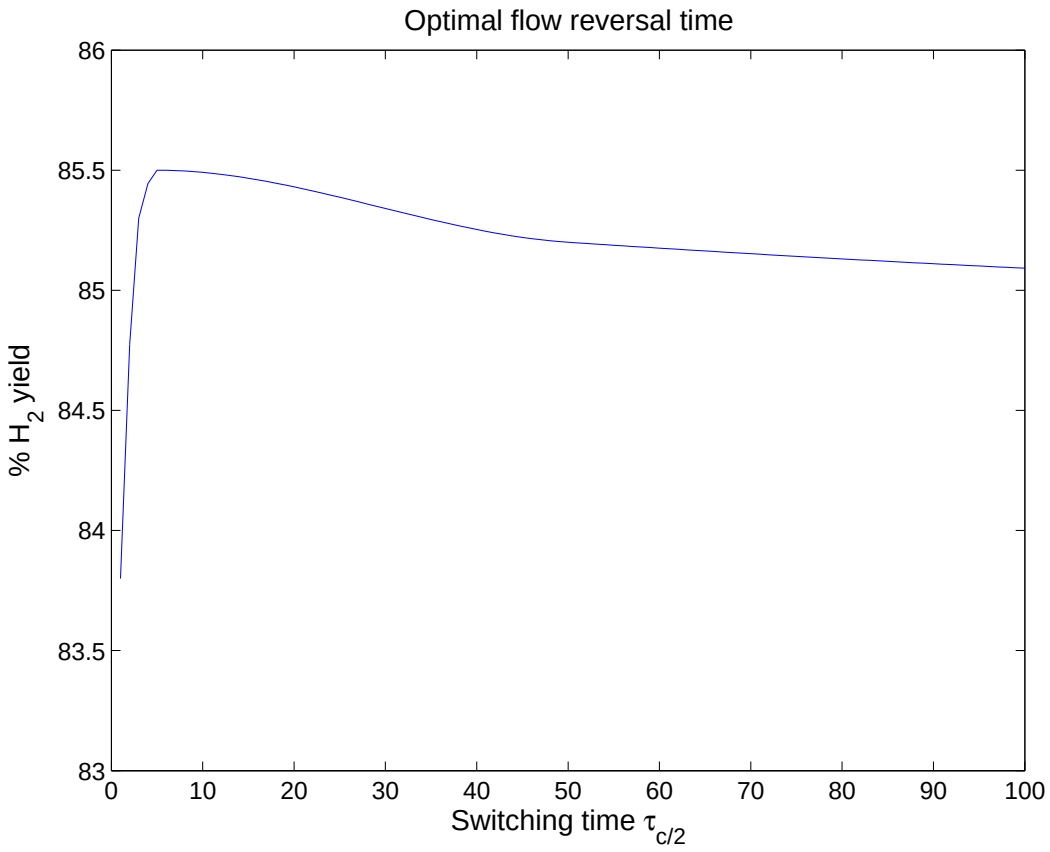

Figure 28: Hydrogen yield as a function of switching time for variations in k_{ref}.

Next, we considered the optimal switching time for the RF operation. Figure 28 shows the hydrogen yield as a function of the switching time $\tau_{c/2}$ for $CH_4 : O_2$ ratio of $1 : 1$. The rate of reforming was decreased by a factor of 0.25 (fourth case in Figure 22). As in the baseline case, the optimal $\tau_{c/2}$ was found to be $5\,sec.$ A similar analysis was performed for couple more variations in the rate constants; each time the optimal switching time was found to be about $4 - 5\,sec.$ Thus, variations in kinetic constants did not have a significant effect on the optimal switching time. These results are important because they demonstrate that the optimal conditions for the RF operation do not vary significantly under modest variations in the reactor parameters. Thus, a RF reactor designed at nominal conditions will yield good performance if the kinetic parameters undergo some changes.

4.3 Improved Catalyst Placement

In the previous section, we observed that decreasing the rate of oxidation or increasing the rates of reforming or water gas shift reactions resulted in an improved hydrogen yield. As various noble metals are known to catalyze methane reforming and partial oxidation at different rates [114], an opportunity exists to optimally place various catalysts along the length of the reactor. In case of the RF operation, catalyst placement gains prominence due to the spatial temperature pattern obtained [57]. Specifically, the RF reactor contains an extended central zone of high temperature flanged by end zones of low temperatures. We propose that higher hydrogen yield can be obtained by carrying out the endothermic reforming reactions at the reactor center and slightly exothermic water gas shift reaction at the reactor exit.

Avci et al. [5] and Ma and Trimm [71] reported simulation and experimental studies respectively, comparing the performance of methane oxidation-reforming in a fixed bed reactor containing Pt/Al_2O_3 as an oxidation and Ni/Al_2O_3 as a reforming catalyst. Both reported that a mixed bed reactor (the two catalysts were mixed in the same catalyst bed) provides better hydrogen yield than a dual bed system (oxidation catalyst placed upstream of the reforming catalyst). In both these cases, the authors concluded that better heat and mass transfer between the two catalysts in a mixed bed configuration was responsible for higher hydrogen yields. In contrast, the gas-solid heat and mass transfer coefficients in a microreactor are large due to the small channel diameters. Hence, it stands to be determined if consecutive placement of catalysts would provide higher yields than using a single catalyst. We also note that the previous authors [5, 71] considered UD fixed bed systems, while this section considers the RF microreactor.

4.3.1 Kinetic Expressions

Ever since Fischer and Trospch [33] reported activity of noble metals towards catalyzing methane reforming reactions, studying the kinetics of methane oxidation, reforming and water gas shift reactions has been a topic of great interest for the last century. While nickel has been predominantly used as an industrial reforming catalyst [91], it tends to form

Table 9: Rate constants and activation energies for oxidation, reforming and water gas shift reactions on various catalysts. Cat1, Cat2 and Cat3 are primarily oxidation, reforming and water gas shift catalysts respectively.

Reaction	Oxidation		Reforming		Shift	
Catalyst	k_0	E_a	k_0	E_a	k_0	E_a
Base	23.06	100.32	76.16	114.12	0.412	38.13
Cat1	55.69	83.32	3.05	114.12	1.648	58.13
Cat2	2.31	100.32	952.0	141.12	32.96	43.13
Cat3	11.53	112.32	180.0	130.12	6.59	38.13

carbon residues at higher temperatures and loses its activity under oxidizing conditions [23] that exist at the reactor inlet. Recent studies on steam/dry reforming of noble metals by Wei and Iglesia [113, 114, 116, 115, 112] indicate these to be good candidates for possible microreactor catalysts [33]. Additionally, Wheeler et al. [117] recently performed kinetic studies on water gas shift reaction on these noble metal catalysts at high temperatures. Based on these sources, we conclude that

- Pd is a good oxidation catalyst (Cat1)
- Ni is a good reforming catalyst (Cat2)
- Ru is a good water gas shift catalyst (Cat3)

We compare the results of placement of these catalysts as compared to the baseline case of Pt or Pt/Rh catalyst used in our system. As our aim is to obtain a rational means to study improved catalyst placement, exact reaction kinetics for each catalyst are not critical. The kinetic rate constants are shown in Table 9 and the procedure used to obtain them is elaborated in Appendix D.

4.3.2 Results

The analysis for improving reactor productivity was done in a step-wise manner. First, we studied the effect of using Ruthenium catalyst at the reactor ends. Next, we studied the

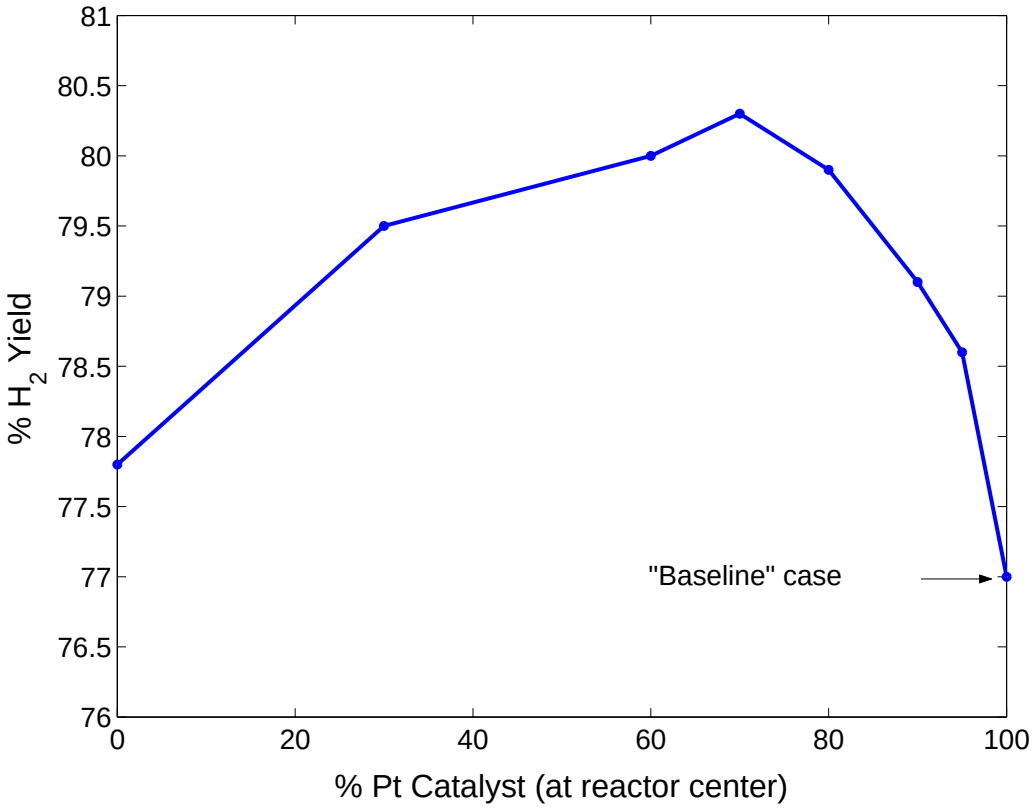

Figure 29: Effect of Ru-Pt-Ru catalyst patterning

effect of patterning Palladium and Nickel catalysts. Finally, we combined these results to obtain an optimal catalyst patterning for the microreactor

- **Patterning of water gas shift catalyst**

  The effect of patterning water gas shift catalyst (Ru) with the nominal Pt catalyst was studied. The catalysts were placed symmetric about the center of the reactor. Figure 29 shows the hydrogen yield obtained as a function of percentage of Pt catalyst at the center for Ru-Pt-Ru configuration. As Ru is a good oxidation and reforming catalyst as well, the performance of pure Ru catalyst is comparable to that of pure Pt catalyst. The optimum hydrogen yield was obtained when Ru catalyst occupied 15% at the two ends of the reactor and Pt occupied the central 70% of the reactor length.

- **Patterning oxidation and reforming catalysts**

We now consider the effect of patterning Pd and Ni catalyst. Pd is an excellent oxidation catalyst, but a very poor reforming catalyst. Conversely, Ni is a poor oxidation catalyst, but a good reforming catalyst. As a result, pure Pd or Ni catalysts give very poor performance. Like Avci et al. [5], we found that a mixed catalyst consisting of Pd/Ni performs better than spatial patterning. For the latter, Pd was placed upstream and Ni placed downstream, resulting in a Pd-Ni-Pd configuration. The Pd/Ni mixed catalyst gave 54.7% hydrogen yield. On the other hand, Pd-Ni-Pd configuration performed very poorly. The maximum hydrogen yield obtained was only 23.7% when Ni occupied the central 80% of the reactor length.

The reason for this can be seen from Figure 30, which shows the temperature profile for a Pd-Ni-Pd catalyst. The oxidation takes place on Pd catalyst resulting in a steep rise in temperature at the initial section of the reactor. However, in the central section, endothermic reforming reaction causes a steep drop in temperature. As little oxidation reaction takes place in this region, the temperature drops significantly. This results in very low reaction rates in the central section of the reactor causing poor conversion of methane.

- **Using reforming catalyst to improve baseline performance**

  Next, we thought that patterning Pt and Ni catalysts might provide significantly better results than patterning Pd and Ni. This is because Pt is a better reforming catalyst than Pd. Figure 31 shows the hydrogen yield as a function of Ni loading in the Pt-Ni-Pt catalyst. The plot clearly shows that pure Pt catalyst performs better than the patterned system.

- **Patterning all catalysts**

  Based on the above results, we learnt two things: 1. Ru catalyst at the reactor ends provides improved hydrogen yields, and 2. pure Pt or Pt/Ni mixed catalyst provides better hydrogen yield than spatially patterned case. Hence we considered the following configuration: [10% Ru]–[80% Pt/Ni mixed]–[10% Ru]. This particular case gave the best hydrogen yield of 86.1% amongst all the cases considered.

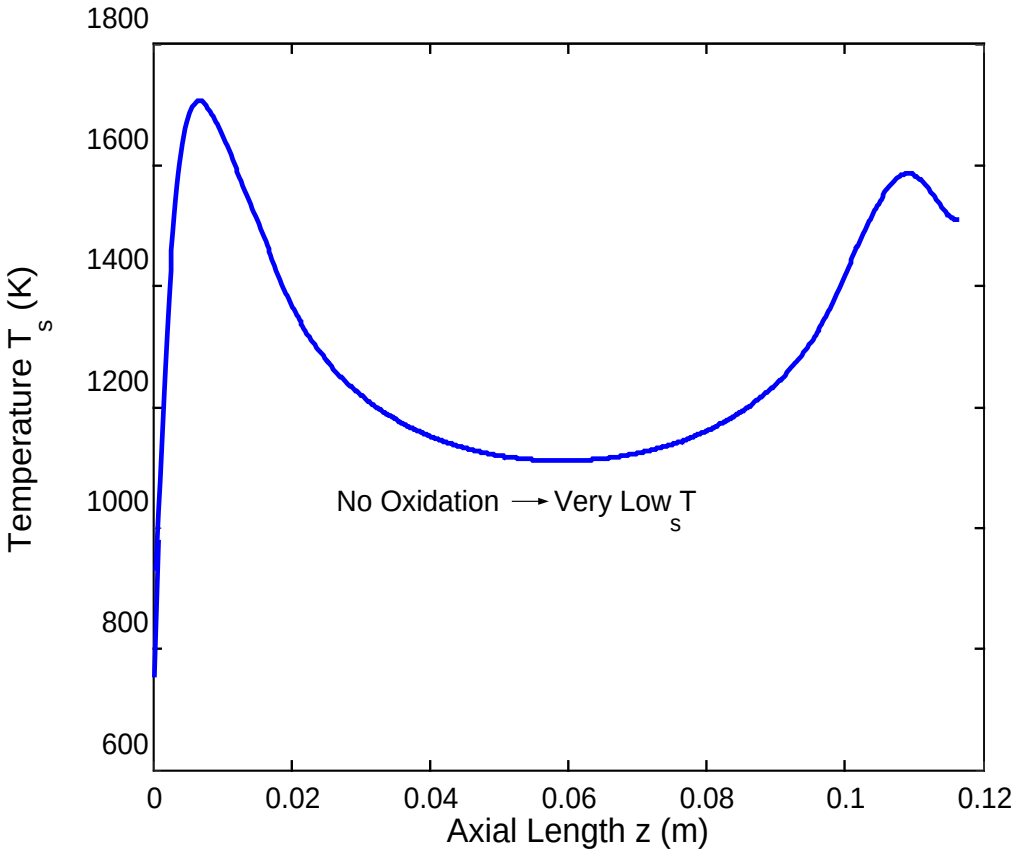

Figure 30: Temperature profile at periodic steady state for [10% Pd]-[80% Ni]-[10% Pd] catalyst patterning

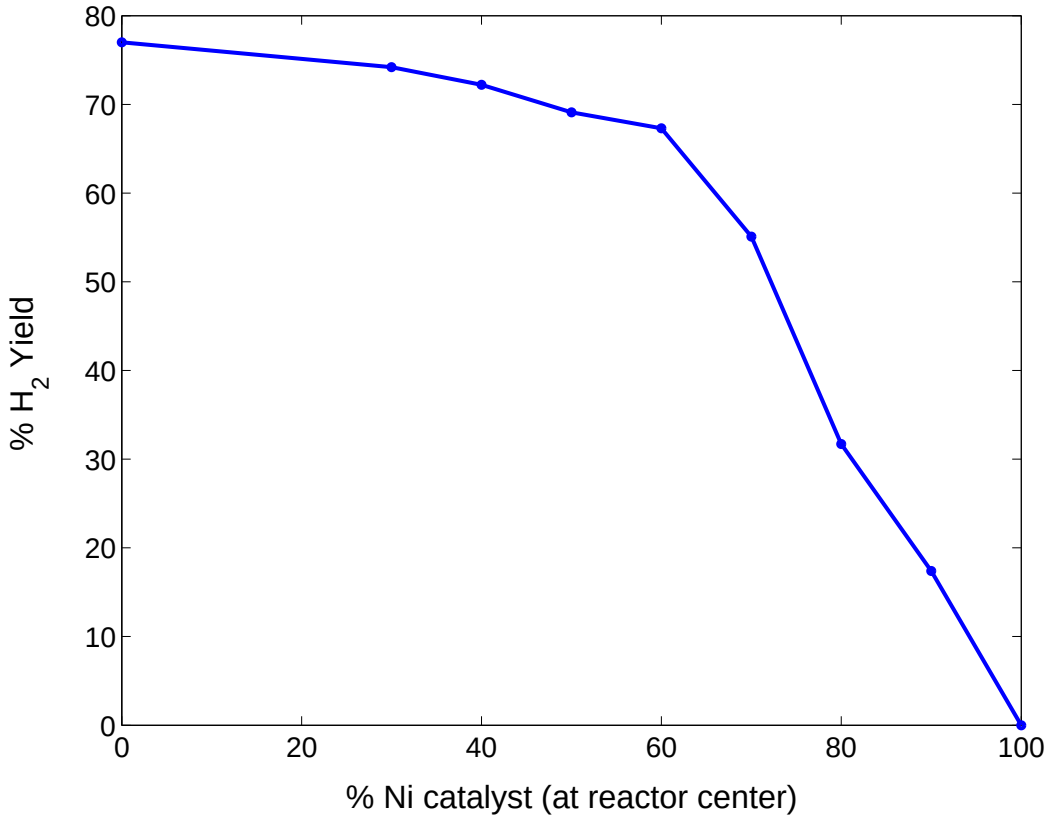

Figure 31: Effect of Pt-Ni-Pt catalyst patterning

4.4 Operability Analysis

The RF operation was shown to be more robust, in that the autothermal operation is maintained under variations in kinetic parameters and feed conditions. The optimal feed ratio was also found to be in the vicinity of $CH_4 : O_2 = 1.15 : 1$. However, the UD operation has an upper limit on the inlet velocity and/or the relative amount of methane in feed, beyond which an autothermal operation is not possible in the UD mode. The operability analysis presented in this section aims to address the following questions:

- What is the maximum hydrogen throughput we can expect from the system under various conditions? (Section 4.4.1)
- How robust is the reactor operation to catalyst deactivation? How much performance do we sacrifice in order to ensure robustness of the reactor operation? (Section 4.4.2)
- What is the effect of increasing catalyst loading or reactor parameters on the reactor performance? (Section 4.4.2)
- What is the effect of properties of the reactor material of construction? (Section 4.4.3)

4.4.1 Hydrogen throughput

The comparison of the UD and RF operations has so far focussed on hydrogen yield, *ie.* the number of hydrogen molecules obtained for every half a molecule of methane input to the reactor (since $CH_4 \equiv 2H_2$). Another important criteria is the throughput of hydrogen, *ie.* milliliters of hydrogen obtained per minute at standard temperature $298\,K$ and pressure $1\,atm$ from the reactor. As the load increases, the hydrogen throughput requirement increases, which then requires higher amount of methane feed into the reactor. We had seen earlier that either increasing the velocity beyond a certain value at constant feed ratio or increasing the relative amount of methane in the feed at constant velocity results in disappearance of the autothermal UD state. For the microreactor, we found that the feed conditions for obtaining maximum hydrogen throughput were inlet velocity $v_0 = 1.75\,m/s$ and $CH_4 : O_2 = 1.15 : 1$, and the maximum throughput was $74.9\,ml\ H_2/min$ at standard conditions. For the same inlet conditions, the hydrogen throughput from the RF operation

is hardly any higher, at 75.1 ml H_2/min. However, autothermal operation can be maintained in the RF mode at much higher inlet velocities; hydrogen throughput well in excess of 100 ml H_2/min can be obtained in the RF operation, if the inlet velocity is increased beyond 2.3 m/s.

4.4.2 Catalyst loading and Catalyst deactivation

Next, we considered the robustness of the UD and RF operations to loss in catalyst activities. The partial oxidation process in the microreactor is kinetically limited. Hence, catalyst deactivation, which is expected to happen over a period of reactor use, will severely affect the reactor operation. In fact, if the kinetic constants were reduced to 25% of the nominal value, no autothermal UD operation was obtained for any relevant feed conditions. Figure 32 shows how the maximum hydrogen throughput decreases with catalyst effectiveness. One can clearly see a substantial decrease in hydrogen throughput with a drop in catalyst effectiveness. The region below the solid line is the region of autothermal UD operation. In order to operate in a region above this curve, one needs to run the reactor in the RF mode.

The nominal design conditions call for operating the UD reactor close to $CH_4 : O_2 = 1.15 : 1$ and $v_0 = 1.75\, m/s$, which provides 74.9 ml/min of hydrogen, with the yield being 75.6%. To give leeway for catalyst deactivation or presence of impurities in the feed, the reactor is designed conservatively to work away from this optimum. We designed the reactor to account for 50% decrease in catalyst activity. Under these conditions, the feed conditions of $CH_4 : O_2 = 1 : 1$ and $v_0 = 1.25\, m/s$ gave the maximum yield. If the reactor is operated at these input conditions to ensure robustness of the UD operation, the hydrogen throughput (for nominal values of kinetic constants) will drop to 64.1 ml/min of hydrogen (down from the optimal value of 74.9 ml/min) and the hydrogen yield will drop to 71.8% (from the optimal value of 77.5%).

Next, increasing the catalyst loading per unit reactor volume was investigated to improve the range of the reactor operation in the UD mode. Increased catalyst loading can be achieved by depositing the catalyst on the walls of the reactor channels and/or by using thin catalyst springs instead of using a straight wire. Assuming that the void volume is

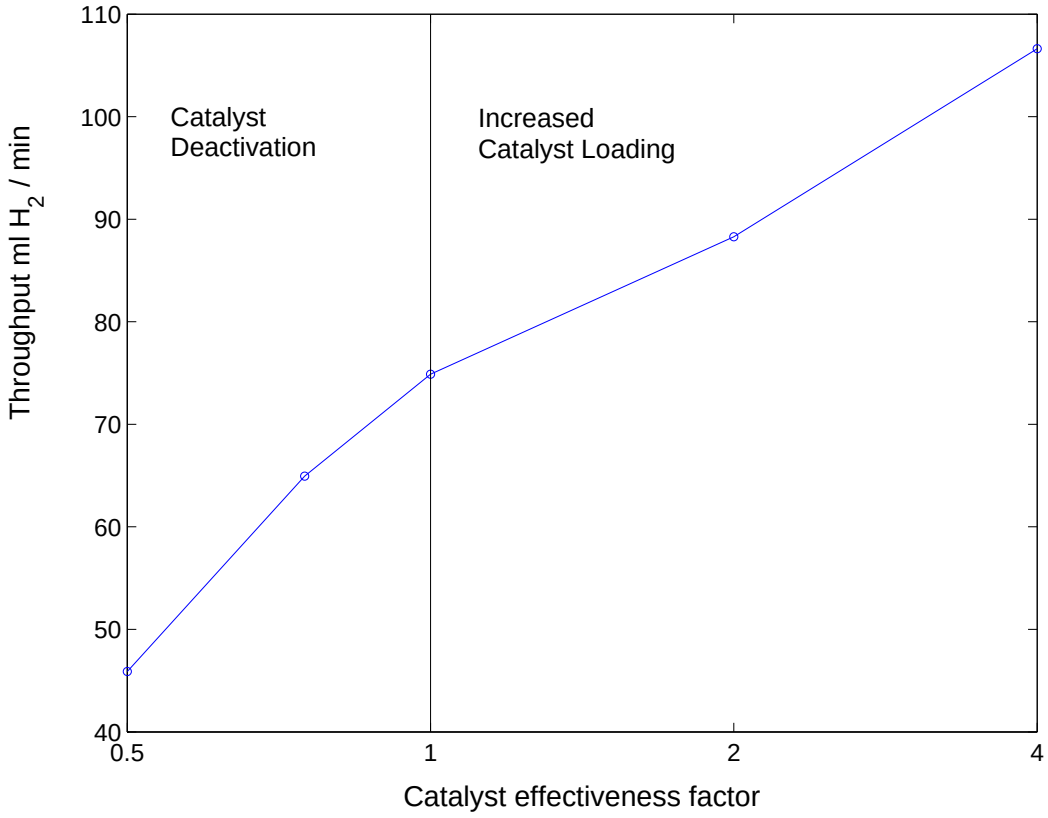

Figure 32: Hydrogen throughput variations with variations in catalyst effectiveness. Left part: catalyst deactivation, right part: increased catalyst loading

not affected, increasing the catalyst loading can be equivalently represented by increasing the kinetic rate constants. As expected, higher hydrogen throughput can be obtained at higher catalyst loading by increasing the inlet velocity and the amount of methane in feed. The right-hand part of Figure 32 shows the effect of increasing the catalyst loading. With a four-fold increase, a hydrogen throughput in excess of $100\,ml\ H_2/min$ can be obtained. In fact, the highest hydrogen throughput for this case is $115ml/min$, with inlet velocity $v_0 = 2.0\,m/s$ and feed $CH_4 : O_2 = 1.5 : 1$. Providing leeway for robust operation, as was done in the previous section[1], a hydrogen throughput of $87.5\,ml/min$ can be obtained, with 79.8% H_2 yield (as opposed to 82.5% H_2 yield at optimal throughput conditions). For the same kinetics, the optimal hydrogen yield in RF operation was 79.5%; at a similar $CH_4 : O_2$ ratio of $1.18 : 1$ as described in section 4.2.2.

Figure 33 shows the operation diagram for the UD reactor operation for various kinetic conditions. The lines in the figure show the maximum velocity that provides an autothermal UD operation as a function of inlet feed ratio, each line representing different kinetic rate constants. The region below and to the left of each line is the region of autothermal UD operation for the corresponding kinetic constant. No autothermal operation is possible for velocities or feed ratios greater than these values. The operating condition that gives maximum hydrogen yield for particular kinetic condition is shown as a diamond. For example, we previously noted that the maximum hydrogen throughput was obteined at nominal operating conditions for feed ratio of $CH_4 : O_2 = 1.15 : 1$ and inlet velocity of $v_0 = 1.75\,m/s$, is indicated as a diamond on the solid line.

4.4.3 Reactor material of construction

We now consider the reactor operation with different materials of construction, such as steel or silicon. Changing the reactor material changes the heat capacity ($\rho_s c_s$) and thermal conductivity (λ_s). We analyzed the effect of varying these two parameters independently.

We found that varying the heat capacity ($\rho_s c_s$) of the reactor did not have any significant effect on the reactor performance. In contrast, thermal conductivity had a very strong

[1] By operating at optimal conditions for catalyst effectivity of 50%

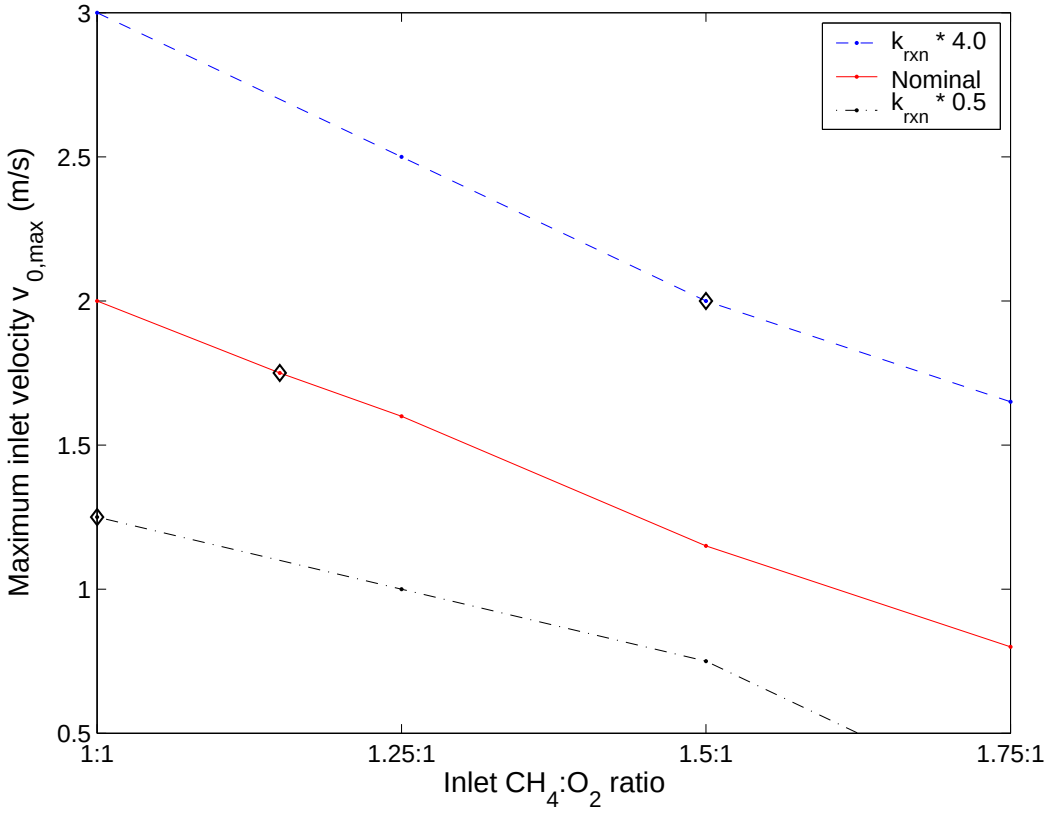

Figure 33: Operation diagram showing the maximum velocity for autothermal UD operation as a function of $CH_4 : O_2$ feed ratio, for various values of kinetic rate constants.

effect on the reactor performance. When λ_s was reduced, the region of autothermal UD operation also reduced. Similarly, increasing λ_s resulted in stabilization of the autothermal UD operation, as well as decreasing the maximum temperature $T_{s,max}$. For example, when λ_s was increased to $12\,W/m^2K$, we obtained autothermal UD operation even with $v_0 = 2.0$ and $CH_4 : O_2 = 1.25 : 1$. This is because higher λ_s results in a higher axial distribution of thermal energy. The maximum temperature is thus reduced, and the temperature at the entrance is increased. Thus, the hot-spot is shifted closer to the entrance, resulting in stabilization of the UD operation. This gain in the operation window comes at the expense of hydrogen yield attainable. For example, at nominal conditions and $CH_4 : O_2 = 1.25 : 1$, the hydrogen yield decreased from 77.0% for $\lambda_s = 6.0\,W/m^2K$ to 74.6% for $\lambda_s = 12.0\,W/m^2K$.

Figure 34 shows the operation diagram for the UD operation for various changes in the solid thermal conductivity, for a feed ratio of $CH_4 : O_2 = 1.25 : 1$. The maximum inlet velocity (solid line, left ordinate) that gives autothermal UD operation and the corresponding hydrogen yields achieved at these operating conditions (dashed line, right ordinate) are plotted as a function of the solid thermal conductivity. The region below the solid line represents the UD operating region, while the one above it represents the region where a travelling thermal wave is formed.

4.5 Opposed Flow reactor

Finally, we considered an alternate "opposed flow" (OF) reactor operation to maintain autothermal steady state without the need of flow reversal. The gases flow in adjacent channels flow in opposite directions in the OF reactor. The incoming cold stream in one channel gets heated due to the hot effluent stream. The reactor itself acts as a regenerative heat exchanger [31] and is analogous in some respects to the RF operation (Figure 21). The main motivation for considering the OF operation is that RF reactor requires complicated valve assembly (such as a 4-way valve) or moving reactor parts to achieve flow reversal. An RF reactor also has large dead volumes during flow reversal, although microreactor designs that minimize dead volume [59] in RF or sequential switching of a reactor network [34] for which dead volumes is not an issue have been proposed. Finally, dynamic analysis and

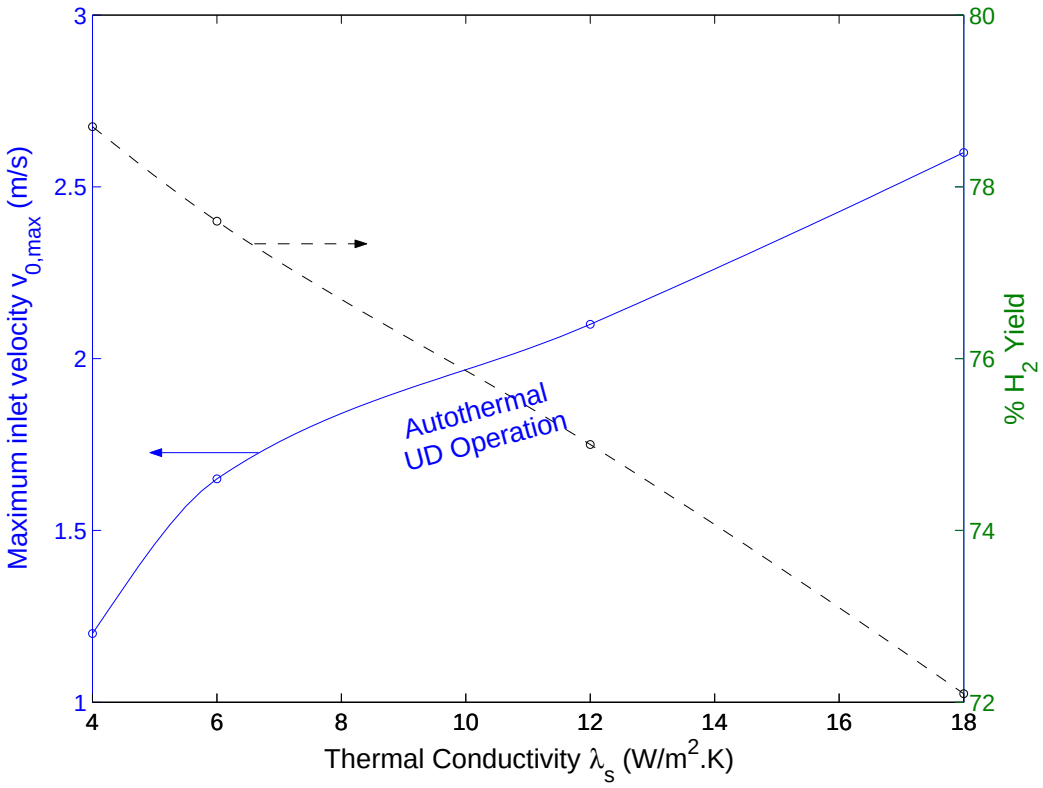

Figure 34: Operation diagram showing the maximum velocity for autothermal UD operation (—) and the corresponding hydrogen yield (– –) for variations in the reactor thermal conductivity. Operating conditions: $CH_4 : O_2 = 1.25 : 1$ and $v_0 = 1.68\,m/s$.

control issues in a RF system are complex due to the repetitive or periodic nature of these processes. On the other hand, the OF reactor usually has stationary (non-periodic) steady state and the control issues are likely to be simpler than a RF reactor.

All the reactor mass and energy balance equations and boundary conditions remain the same as discussed in [56] except an additional channel-channel heat transfer term gets added to the solid heat balance. We note that the two adjacent channels are alike in all respect, giving symmetry to the system. If we represent any state variable in the two adjacent channels as ϕ_1 and ϕ_2, the symmetry results in

$$\phi_1(z,t) = \phi_2(l-z,t) \tag{17}$$

The thermal interaction between the two channels is accounted in the energy balance equation as

$$\begin{aligned} \frac{\partial}{\partial t}T_s(z) &= \frac{\lambda_s}{\rho_s c_s}\frac{\partial^2}{\partial z^2}T_s(z) - \frac{h_v \underline{a}}{\rho_s c_s}\left[T_s(z) - T_g(z)\right] - \\ &\quad \frac{h_{c-c}\underline{a}}{\rho_s c_s}\left[T_s(z) - T_s(l-z)\right] \end{aligned} \tag{18}$$

The only additional term compared to the previous model is the channel-to-channel heat transfer term $h_{c-c}\left[T_s(z) - T_s(l-z)\right]$. An alternative way to model this system is to consider separate sets of balance equations for the opposing flow directions. The channel-to-channel heat transfer term in this case becomes $h_{c-c}\left[T_{1s}(z) - T_{2s}(z)\right]$.

We consider the performance of OF reactor for various values of the channel-to-channel heat transfer coefficient h_{c-c}, the time constant which is given by

$$\tau_{c-c} = \frac{\rho_s c_s}{h_{c-c}\underline{a}} \tag{19}$$

Figure 35 shows the hydrogen yields for the baseline case for various changes in the time constant τ_{c-c}. The hydrogen yield for the OF operation is slightly better than the RF operation. Additionally, the hydrogen yield does not vary significantly for comparatively large changes in the channel-channel heat transfer coefficient. The temperature profiles in the OF and RF operations are shown in Figure 36. Qualitatively, the temperature profiles in the two cases are similar; *ie.*, there is an extended region of high temperature at the

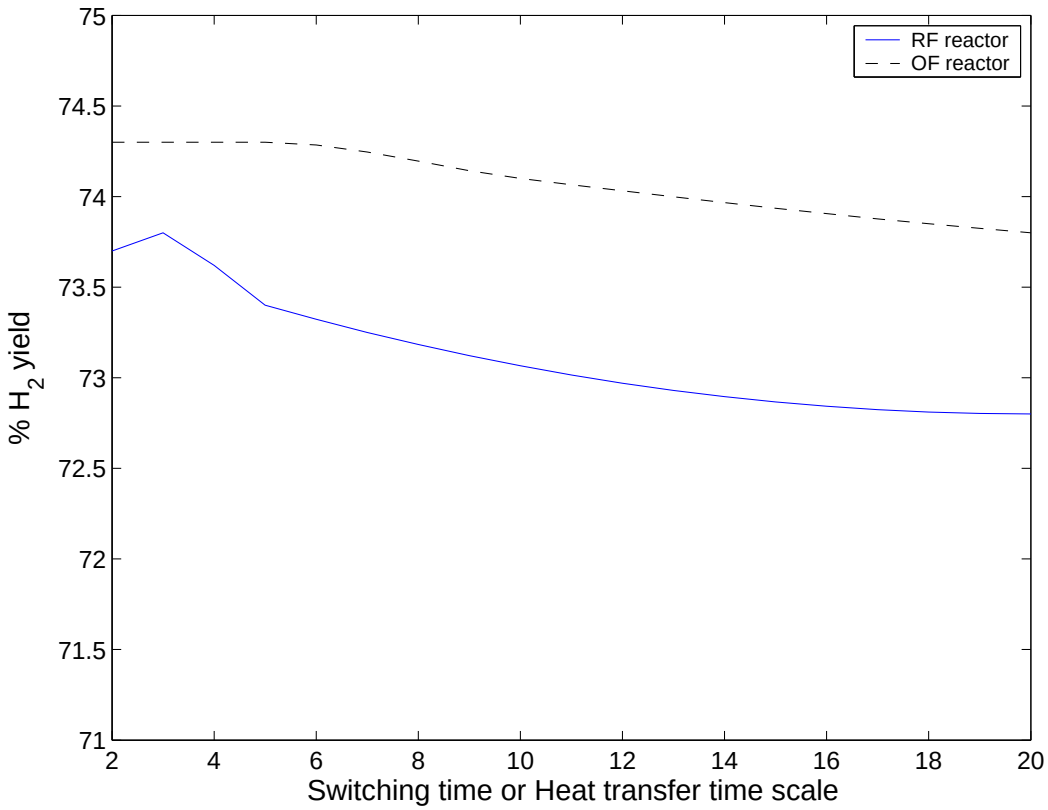

Figure 35: Hydrogen yield as a function of time scale of channel-to-channel heat transfer τ_{c-c} for the OF reactor and switching time $\tau_{c/2}$ for the RF reactor.

reactor center flanked by regions of comparatively lower temperatures at the reactor ends. The maximum temperature in the OF operation was higher than the RF operation. Thus the OF operation therefore is similar to the RF operation under the limit of high frequency of flow reversal.

Finally, we evaluated the performance of the OF operation under conditions simulating catalyst deactivation, for a range of input $CH_4 : O_2$ ratios. We found that autothermal operation was maintained in the OF operation for higher hydrogen throughput requirements, even for lower reaction rates. However, the optimal feed ratio changed for various changes in the kinetic constants. For example, the optimal feed ratio was $CH_4 : O_2 = 1.23 : 1$ for the nominal case, while it changed to $1.18 : 1$ when the kinetic constants were reduced by half. The change in the optimal feed ratio, however, is not as significant as observed in

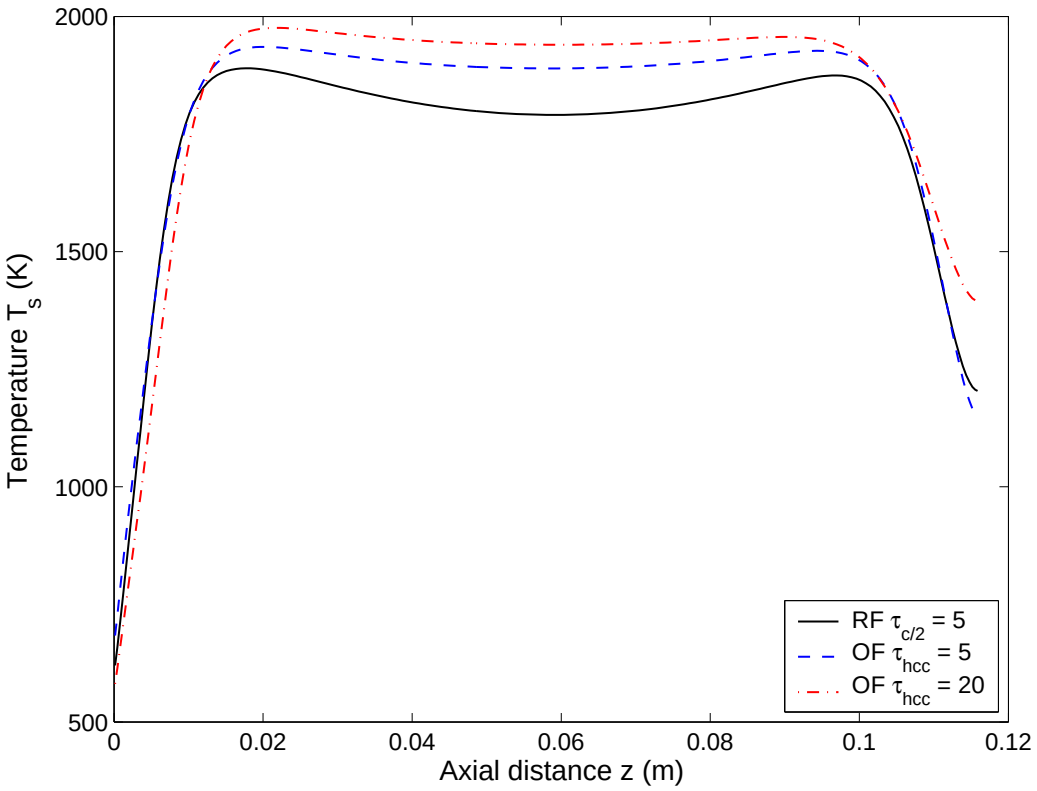

Figure 36: Comparison of the temperature profiles in the RF operation with that in the OF operation for various time scales of channel-to-channel heat transfer τ_{c-c}.

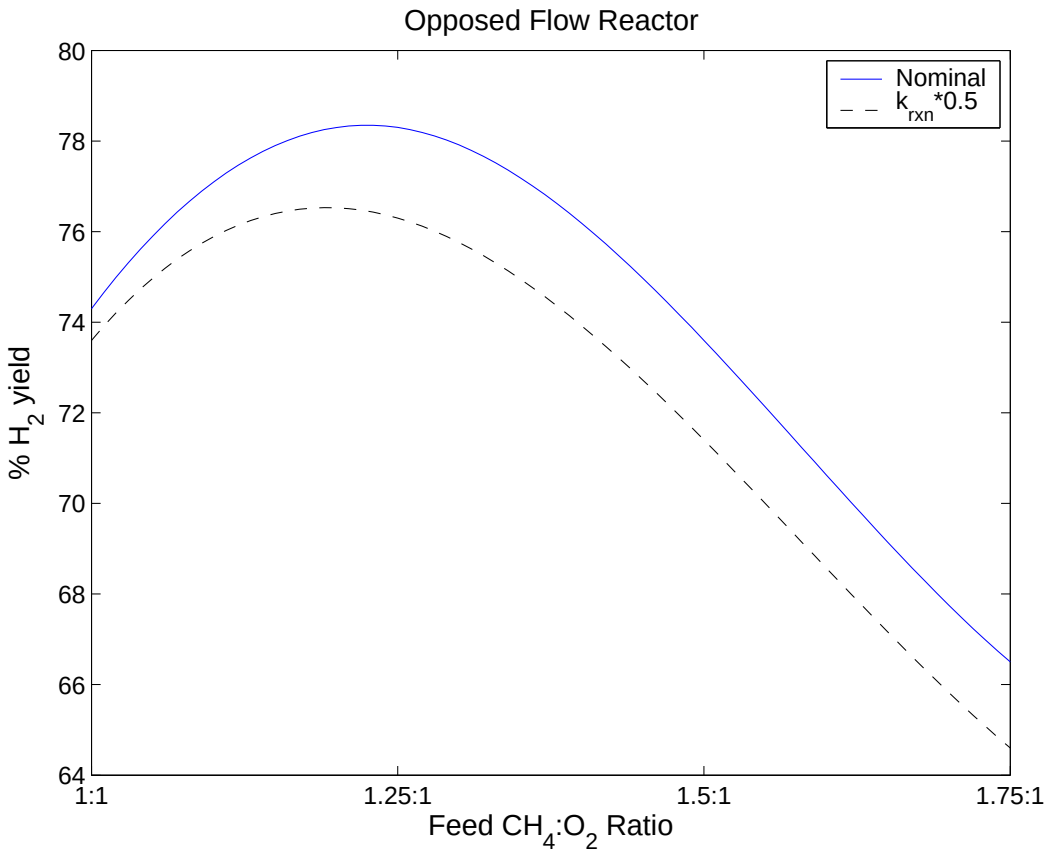

Figure 37: Hydrogen yield as a function of inlet feed ratio for OF operation for two different reaction kinetics.

the UD operation. The opposed flow reactor therefore retained the simplicity of the unidirectional operation, while providing higher hydrogen throughput and robustness of the reverse-flow operation. Fabrication of the OF reactor is not significantly difficult compared to an UD reactor, as displayed in Figure 38-a. Often, increased throughput is obtained by stacking these planar MEMS assemblies, such as the one in Figure 38-b, which is based on the periodic microreactor of Rouge et al. [92].

4.6 Conclusions

A sensitivity and operability analysis of the unidirectional (UD) and reverse-flow (RF) operations of methane partial oxidation in a microreactor was presented in this report. The reactor operation was found most sensitive to the rate constants of oxidation and reforming reactions, and to the thermal conductivity of the reactor material. We found that the

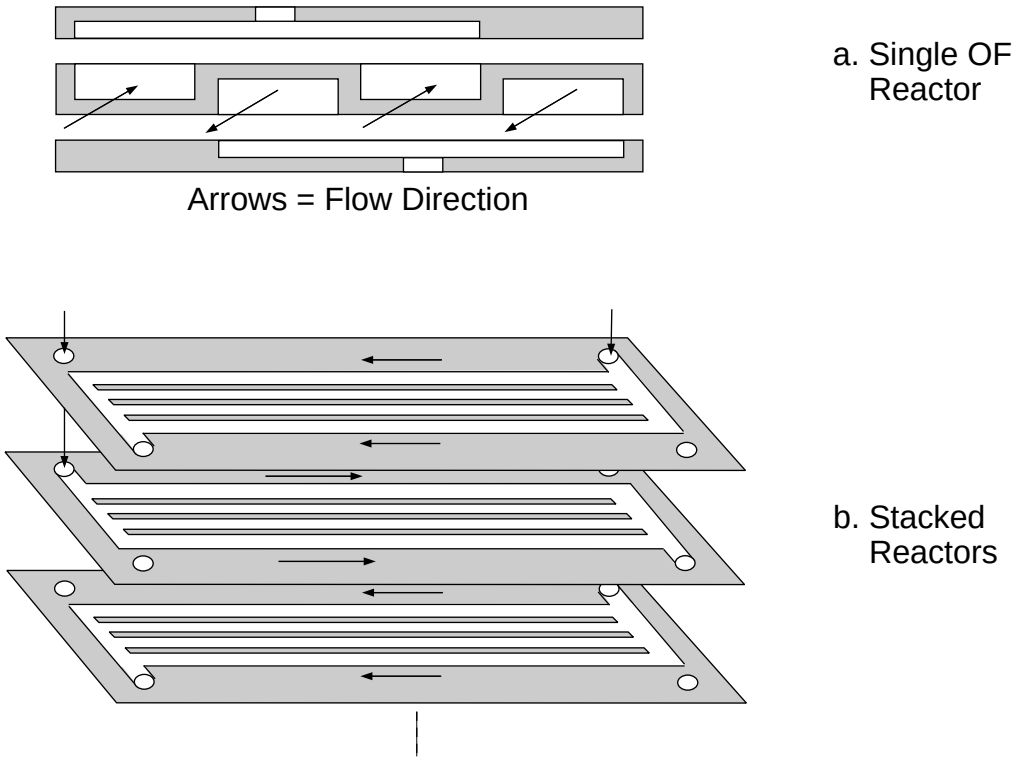

Figure 38: Possible designs of a single MEMS opposed flow reactor and an assembly consisting of a reactor stack (adapted from [92]), with arrows indicating the flow directions.

optimal feed ratio varied significantly with variations in the kinetic rate constants for the UD operation. In contrast, it remained more or less constant at about $CH_4 : O_2 = 1.15 : 1$ for the RF operation.

There was an upper limit on the inlet velocity and the relative methane content in the feed for which an autothermal operation could be maintained in the UD mode. The range of conditions for autothermal operation increased with an increase in catalyst loading and and increase in thermal conductivity of the reactor. Similarly, the range of autothermal operation reduced when catalyst deactivation was accounted for maintaining robust operation. Based on this, operation diagrams were developed for the UD mode, expressing the maximum inlet velocity as a function of inlet feed ratio, and thermal conductivity of the reactor. In contrast, we did not observe an upper limit on the velocity for autothermal RF operation (hence pressure drop, maximum temperature and other criteria would determine the maximum velocity in the RF operation).

An improved catalyst placement scheme based on the temperature profiles in the RF operation was also suggested. We found that placing water gas shift catalyst like Ruthenium at the reactor ends and a mixed oxidation-reforming catalyst of Platinum and Nickel at the center gives the most optimal performance amongst the noble metal catalysts investigated.

Finally, an opposed flow scheme was suggested to maintain autothermal operation without the complicated flow reversal. In the opposed flow reactor, gas flows in opposite directions in adjacent channels (like a counter-current heat exchanger). This reactor behaves like the RF reactor under the limit of infinitely fast port switching.

PART II

Switching Between Multiple Steady States

CHAPTER V

CYBERNETIC MODEL PREDICTIVE CONTROL OF BIOREACTORS WITH MULTIPLE STEADY STATES

The control of bioreactors displaying steady state multiplicity is considered in this chapter. Steady state multiplicity occurs due to the ability of microbial and mammalian cultures to regulate their metabolic pathways in response to the environment. Cybernetic modeling framework is used to capture these metabolic regulations associated with switching between multiple steady states. Two different examples, viz. a microbial reactor and a mammalian cell culture are considered. Preliminary results on optimal switching between metabolic states of the bioreactors using sequential linearization-based Model Predictive Control (slMPC) are presented.

5.1 Background and Motivation

Multiplicity of steady states is a condition in which a system displays two or more distinct states and output conditions for the same set of input conditions. A classical example of this phenomenon is seen in a non-isothermal stirred tank reactor with exothermic reaction [4, 37]. It is a subject of great interest to chemical engineers and has been observed in many process systems, such as distillation [44], polymerization reactors [109], fixed-bed reactors [55], enzymatic reactors [21] etc. Steady state multiplicity arises due to the nonlinearity of the chemical process. For example, in a non-isothermal reactor with exothermic reaction, the conversion of reactants has a nonlinear dependence on temperature; in turn, the conversion affects the rate of heat release and consequently the temperature. At low temperature, the conversion of reactants is lower and the amount of heat generated balances the amount of heat loss. For the same inlet conditions, another steady state with higher temperature may coexist: the higher conversion results in higher heat generation, which balances the heat loss (which also depends on temperature).

Multiplicity of steady states in continuous bioreactors — which are operated isothermally — arises from the nonlinearity of the governing kinetics. For a simple unstructured growth model, the necessity of substrate or product inhibition kinetics to obtain steady state multiplicity was demonstrated theoretically by Agarwal et al. [2] and experimentally by Lei et al. [70]. However, the real source of steady state multiplicity in continuous bioreactors lies in the tendency of living cells to regulate metabolic pathways in response to environmental pressures [81]. The different steady states have cells with widely varying metabolic activities. The metabolic activities in the different pathways determine the rates of substrate uptake, cell growth and product formation. A striking example of such steady state multiplicity was observed in hybridoma cell cultures [36, 52] for producing monoclonal antibodies. Three distinct steady states with different concentrations of cell and antibodies were obtained [52].

The difficulty in controlling systems with steady state multiplicity lies in their inherent nonlinear nature. Often, the desired steady state may have only a limited region of attraction and/or the dynamic response of these system in two steady states may be very different. For example, Hernández and Arkun [46] considered control of a stirred tank reactor displaying steady state multiplicity at the unstable steady state. A linear model was insufficient to capture the reactor dynamics and a nonlinear ARMA model was identified and used instead. The necessity of nonlinear models in controlling reactive distillation columns with multiple steady states has also been demonstrated [98]. The authors report that controlled perturbations or manipulating column material balance were ineffective in steady state transition. Steady state transition can only be obtained, conceptually, by a "catastrophic shift" meaning that the manipulated variable (MV) is moved to a value beyond the range of multiplicity to cause the transition and brought back to the nominal value. However, this method was infeasible in practice as it lead to destabilization of the distillation column. On the other hand, Jacobsen and Skogestad [53] reported that tight feedback control of either tray temperature or a single product composition was required to stabilize the unstable operation of a distillation column. The open loop instability, specifically the unstable poles and zeros, determine whether closed loop performance is achievable

or not.

In addition to the nonlinear dynamics, the long-lived transient response, especially in the case of metabolic switches in bioreactors, provides another control challenge. In other words, the trajectories of system response to the input manipulations are flat for large periods of time, followed by a relatively short period when the steady state switch actually occurs. Physically, the cellular regulatory processes change slowly in response to the environment: key enzymes catalyzing certain reactions are produced and activated, while some other enzymes are repressed. This is a sluggish process, following which the system responds rapidly and the metabolic switch takes place. Such long range phenomena require a longer prediction horizon for control, resulting in increased computational complexity. Another issue is relates to the model, which could become discontinuous or non-differentiable at the "boundaries" where the switching occurs.

In this chapter, the optimal switching of a well-mixed bioreactor from an undesirable low yield state to the desirable high yield state is studied. Two different systems are considered: a bacterial culture growing on two substitutable substrates and a mammalian hybridoma culture. The system is modeled using the cybernetic modeling framework [102, 111], which is described in section 5.2. A Model Predictive Control (MPC) scheme based on successive linear approximations of the model [65] at each time in the prediction horizon is presented in section 5.3. Simulation results of applying this successive linearization-based MPC (slMPC) to the bacterial and hybridoma cultures are presented in section 5.4. Finally, general conclusions and remarks are presented in the final section of this chapter.

5.2 Reactor Modeling using Cybernetic Framework

Continuous bioreactors often display steady state multiplicity, and significant delayed responses to changes in the environment due to the tendency of living cells to switch metabolic states in response to environmental pressures. A model framework that captures mechanisms of cellular regulation is required to numerically reproduce the observed steady state multiplicity. The cybernetic modeling framework of Ramkrishna and coworkers [102, 111] is specifically designed to address the metabolic regulations in cells. The model postulates

that cells have learnt to utilize the limited resources available to them in some kind of an optimal way. Thus, when faced with a nutritional choice, the cells regulate metabolic pathways by altering the enzyme synthesis rate and by modulating the enzyme activity. This regulation of enzyme synthesis and activity is governed by the so-called *cybernetic regulation variables* u_i and v_i respectively.

The first step in obtaining a model is abstraction of the complex metabolic network into a small set of key metabolic processes and intermediates. These metabolic processes are catalyzed by key enzymes, which represent an "investment" of the limited cellular resources (such as amino acids, carbohydrates, energy, cofactors etc). At a local level, cell modulates enzyme production and activity between several processes that compete for the same shared resources so as to maximize the products of the competing metabolic pathways. This is done through the cybernetic regulation variables u_i and v_i. For example, an unmodified rate of i^{th} metabolic reaction catalyzed by key enzyme E_i is given by

$$r_i = \mu_i^{max} \frac{s_i}{K_i + s_i} \frac{e_i}{e_i^{max}} \tag{20}$$

The rate of formation of the enzyme is similarly given by

$$r_{e,i} = k_{e,i} \frac{s_i}{K_{e,i} + s_i} \tag{21}$$

Here, we have considered the simplest Monod-type kinetics for the key metabolic reaction. The rate of metabolic reaction and enzyme production are modulated by the cybernetic regulation variables as:

$$\text{net rate} = r_i v_i \qquad \text{enzyme production rate} = r_{e,i} u_i \tag{22}$$

If the aim of the metabolic competition is to maximize the rate of product formation, these cybernetic variables take the form [111]:

$$u_i = \frac{r_i}{\sum_j r_j} \qquad v_i = \frac{r_i}{\max_j r_j} \tag{23}$$

This forms a local control mechanism in cells. A global feedback mechanism also exists, which modulates metabolic rates through cofactors, coenzymes, ATP molecules, etc. The knowledge of such global interactions is also incorporated in writing the balance equations

through the use of global cybernetic variables, in addition to the local cybernetic variables described above. A metabolic switch is governed by these cybernetic regulation variables. From Eq. (23), one can see that both u_i and v_i take a value between 0 and 1, 0 representing completely deactivated state and 1 representing completely activated state. A metabolic switch is often associated with a large change in the values of some of the cybernetic regulation variables. Especially, the variable v_i often becomes non-differentiable (i.e. the first derivative is discontinuous) during metabolic switches; wherein one or more key enzymes go from fully active state ($v_i = 1$) to a partially active state ($v_i = r_i/r_{sup}$) or vice versa.

The cybernetic principles described above were employed to obtain models for a microbial and a mammalian system, as discussed in the remainder of this section.

5.2.1 Bacterial growth on two substrates

The first example considered is a continuous stirred tank containing bacterial cells such as *Klebsiella oxytoca* growing on a mixture of two substitutable nutrients: glucose and arabinose. This system was originally studied for a batch reactor by Kompala et al.[62] for modeling the diauxic behavior of the system. The model utilized data obtained from growth of the bacterium on single substrates and did not require *a priori* specification of the order in which the substrates are consumed. This model does not account for maintenance phenomenon occurring at low dilution rates; it simulates continuous bioreactors at dilution rates close to the maximum growth rates.

The metabolic pathway for cell growth is condensed into a single reaction representing the assimilation of each substrate S_i by biomass C. This reaction is catalyzed by the enzyme E_i, which is induced in the presence of S_i. The limited cellular resources are utilized in an optimal manner by regulating the fluxes through the two competing pathways, as shown in Figure 39-(a). The five states of the system correspond to substrate concentrations (s_1, s_2), biomass concentration (c) and concentration of the two key enzymes (e_1, e_2) within the cells. The capital case symbols represent specific species and the corresponding lower case symbols represent their concentrations. Table 10 shows key variables and parameters for

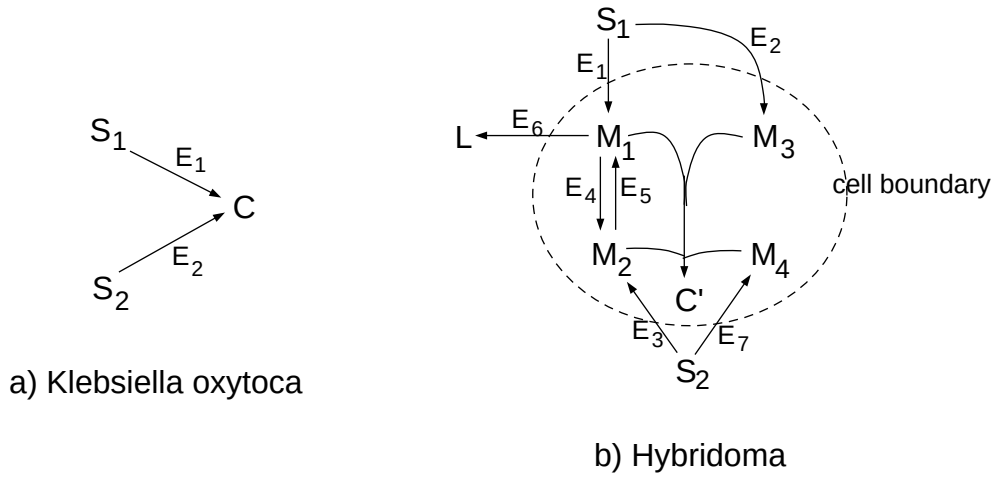

Figure 39: The competing metabolic pathways in *Klebsiella oxytoca* (a) and the abstracted metabolic network of hybridoma (b)

this system. The model consists of five ODEs:

$$\frac{ds_1}{dt} = D[s_{1f} - s_1] - Y_1[r_1 v_1]c \tag{24}$$

$$\frac{ds_2}{dt} = D[s_{2f} - s_2] - Y_2[r_2 v_2]c \tag{25}$$

$$\frac{de_1}{dt} = r_{e1}u_1 + r^*_{e1} - \beta_1 e_1 - r_g e_1 \tag{26}$$

$$\frac{de_2}{dt} = r_{e2}u_2 + r^*_{e2} - \beta_2 e_2 - r_g e_2 \tag{27}$$

$$\frac{dc}{dt} = r_g c - Dc \tag{28}$$

where the rates are given by

$$r_i = r_i^{max}\frac{s_i}{K_i + s_i}\left(\frac{e_i}{e_i^{max}}\right)$$

$$r_{ei} = \alpha_i \frac{s_i}{K_{ei} + s_i}$$

$$r_g = r_1 v_1 + r_2 v_2$$

As seen in equations (24) and (25), the Monod type kinetics are modified by cybernetic regulation variables of the second type v_i, that modify enzyme activity. Similarly, the cybernetic regulation variables u_i in equations (26) and (27) modify the rates of enzyme

Table 10: Key variables and parameters of the system

State Variables	s_1	glucose (gm/L)
	s_2	arabinose (gm/L)
	e_1	key enzyme(s)-1 (gm/gm dry wt.)
	e_2	key enzyme(s)-2 (gm/gm dry wt.)
	c	biomass (gm/L)
Manipulated Variable	D	dilution rate (hr^{-1})
Controlled Variable	c	biomass
Parameter	s_{2f}	s_2 feed rate (gm/L)

synthesis. Mathematically, the cybernetic regulation functions are related to the reaction rates as follows:

$$u_i = \frac{r_i}{r_1 + r_2} \qquad v_i = \frac{r_i}{\max(r_1, r_2)}$$

Rate constants and model parameters are given in Table 11

Table 11: Rate constants and model parameters for the bacterial system

r_1^{max}	K_1	α_1	β_1	K_{e1}	$(Y_1)^{-1}$	r_{e1}^*
1.08	0.01	0.001	0.05	0.01	0.52	10^{-6}
r_2^{max}	K_2	α_2	β_2	K_{e2}	$(Y_2)^{-1}$	r_{e2}^*
0.94	0.01	0.001	0.05	0.01	0.52	10^{-6}

Numerical bifurcation analysis of the above-mentioned cybernetic model of bacterial growth on substitutable substrates revealed the existence of two stable steady states in a certain range of operating parameters [81], which arise due to cells' ability to switch their physiological states under nutritional pressures. In fact, five steady states are predicted by the bifurcation analysis, two of which are stable. Figure 40 shows the bifurcation diagram of steady state concentration of s_1 at a dilution rate of $0.8\,hour^{-1}$ for various values of the relative concentration of S_1 in feed. The steady state which results in high biomass yield is the desired state. Values of the state variables for the two different steady states are shown in Table 12. One can observe that the "working steady state" is close to turning

point bifurcation. Thus, relatively small changes in dilution rate and/or substrate feed concentrations could cause the reactor to drift to the other steady state.

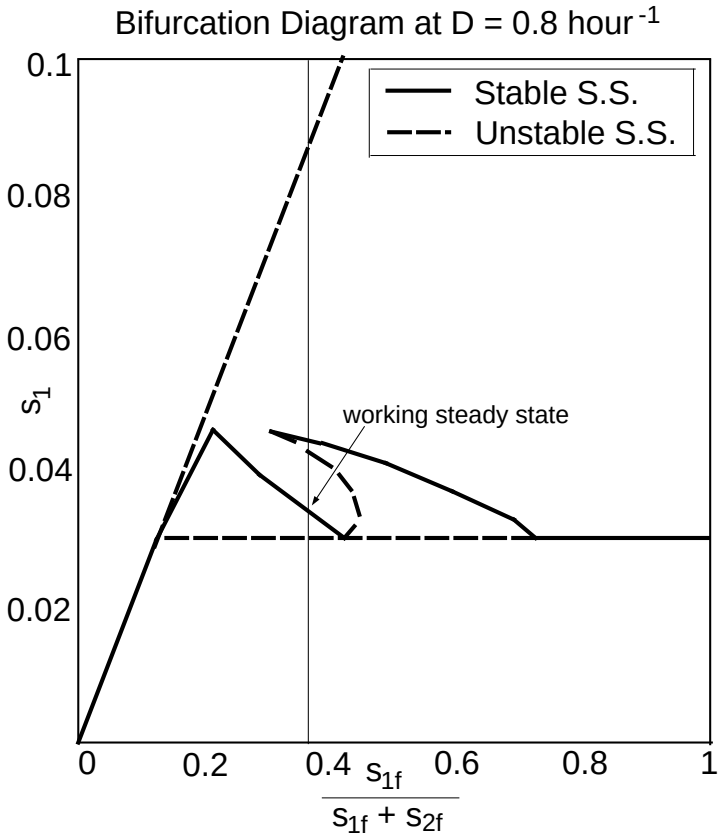

Figure 40: Steady state bifurcation diagram for *Klebsiella Oxytoca* growing on glucose and arabinose. Note the proximity of the steady states to turning point bifurcation. Adapted from [81]

5.2.2 Continuous Hybridoma culture

The second example is of a continuous culture of hybridoma cells. The hybridoma cells considered in this study are hybrids of monoclonal antibody (MAb) producing mammalian B-lymphocyte cells and myeloma cells. Glucose and glutamine are the chief nutrients for hybridoma cells. The desired products of interest are monoclonal antibodies, which are secreted within the cells and are therefore not modeled separately. Hu and coworkers [52, 32] reported that carefully starving the cells in a continuous culture resulted in higher antibody yield. On the other hand, if the culture was not carefully controlled, the mammalian cells

Table 12: Steady state values for input conditions $D = 0.8, s_{1f} = 0.078, s_{2f} = 0.146$.

State	s_1	s_2	e_1	e_2	c
High biomass	0.035	0.081	0.0004	0.0006	0.0565
Low biomass	0.0447	0.1425	0.0007	0.0003	0.02

tended to produce larger amounts of lactate and other byproducts and lower amounts of antibodies. Three different states with low, intermediate and high viable cell mass concentration were reported for the same set of input conditions [32]. This steady state multiplicity was reproduced numerically by Namjoshi et al. [79]. We use an older version of the hybridoma model [NAMJOSHI, A. A., personal communication] in this work. Fig. 41 shows the three distinct steady states observed experimentally in the continuous hybridoma cultures. MAbs are not secreted by the cells, but remain within the cells. Hence, the amount of MAbs is directly proportional to the concentration of viable cells in the reactor.

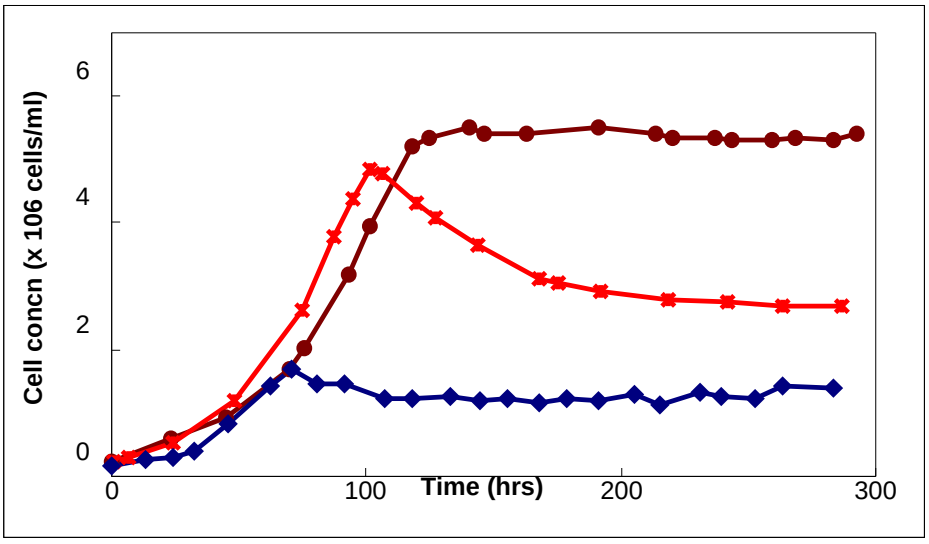

Figure 41: Experimentally observed multiple steady states in a hybridoma reactor [32]

Figure 39-b shows the metabolic network of the hybridoma cells. Glucose (S_1) and glutamine (S_2) are the main nutrients for the cells. Glucose enters the glycolytic pathway to form pyruvate (M_1), as well as enters the pentose phosphate pathway (M_3). Glutamine provides the essential amino acid precursors (M_4) that glucose cannot provide. Both glucose

and glutamine can provide energy requirements of the cell by entering the TCA cycle (M_2 represents TCA cycle intermediates). Lactate (L) is formed as a waste byproduct of the process. Glucose, glutamine and lactate are species external to the cell. They are expressed in terms of gm/l and their balances are given by:

$$\frac{ds_1}{dt} = D\left(s_{1f} - s_1\right) - \left[r_1 v_1^{sc} v_1^{sd} + r_1^m + r_2 v_2^{sd} + r_2^m\right] c \tag{29}$$

$$\frac{ds_2}{dt} = D\left(s_{2f} - s_2\right) - \left[r_3 v_3^{sc} v_3^{sd} + r_3^m + r_7 v_7^{sd} + r_7^m\right] c \tag{30}$$

$$\frac{dl}{dt} = D(-l) + y_6\left[r_6 v_1^{sd}\right] c \tag{31}$$

As in the previous model, the Monod-type reaction rates are modified by the cybernetic regulation variables v_i. The term r_1^m is a maintenance term (m superscript stands for maintenance) for the glycolysis pathway. The maintenance term ensures that flux in all essential pathways remain non-zero and that they are never "turned off."

The rate expressions in Eq. (29–31) are as follows

$$r_i = r_i^{max} \frac{\xi}{K_i + \xi} \frac{c'}{K_g' + c'} \left(\frac{e_i}{e_i^{max}}\right) \tag{32}$$

$$r_i^m = r_i^{m,max} \frac{\xi}{K_{im} + \xi} \frac{c'}{K_g' + c'} \tag{33}$$

where $\xi = s_1$ when $i = 1, 2$; s_2 when $i = 3, 7$; m_1 when $i = 4, 6$; and m_2 when $i = 5$.

The growth intermediates M_i are internal to the cell. Their concentrations m_i are expressed as grams per gram dry weight of biomass. The rest of the cellular mass, in addition to these intermediates is clubbed together as C'. Since these internal species are normalized with respect to the total cell mass, they satisfy the relation $\sum_i m_i + c' = 1$. The balances for these cellular intermediates are given by:

$$\frac{dm_1}{dt} = y_1\left[r_1 v_1^{sc} v_1^{sd} + r_1^m\right] - r_6 v_1^{sd} - r_4 v_4^{sc} + y_5 r_5 v_5^{sc} - y_{m_1 g} r_g' - r_g m_1 \tag{34}$$

$$\frac{dm_2}{dt} = y_3\left[r_3 v_3^{sc} v_3^{sd} + r_3^m\right] + y_4 r_4 v_4^{sc} - r_5 v_5^{sc} - y_{m_2 g} r_g' - r_g m_2 \tag{35}$$

$$\frac{dm_3}{dt} = y_2\left[r_2 v_2^{sd} + r_2^m\right] - y_{m_3 g} r_g' - r_g m_3 \tag{36}$$

$$\frac{dm_4}{dt} = y_7\left[r_7 v_7^{sd} + r_7^m\right] - y_{m_4 g} r_g' - r_g m_4 \tag{37}$$

$$\frac{dc'}{dt} = r_g' - r_g c' \tag{38}$$

where

$$r_g' = r_g'^{max} \prod_{i=1}^{4} \frac{m_i}{K_g + m_i} \tag{39}$$

and the growth rate r_g is the sum of rates of formation and consumption of all internal cellular species.

Finally, the balance for cell mass c is straightforward

$$\frac{dc}{dt} = r_g c - Dc \tag{40}$$

Enzyme Balances: for all the key enzymes are given by

$$\frac{de_i}{dt} = r_{e_i} u_i + r_{e_i}^* - \beta_i e_i - r_g e_i \qquad \forall i = 1, 3 - -7 \tag{41}$$

The significance of each term in the above balance is same as in the previous case. Enzyme balance is required for all enzymes except E_2. This is because in glucose limiting situations seen in the system, the pentose phosphate pathway is always saturated.

Cybernetic Variables:

$$\begin{aligned} u_1^{sd} &= \frac{r_1}{r_1 + r_2} & v_1^{sd} &= \frac{r_1}{\max(r_1, r_2)} \\ u_2^{sd} &= \frac{r_2}{r_1 + r_2} & v_2^{sd} &= \frac{r_2}{\max(r_1, r_2)} \\ u_1^{sc} &= \frac{y_1 r_1}{y_1 r_1 + y_5 r_5} & v_1^{sc} &= \frac{y_1 r_1}{\max(y_1 r_1, y_5 r_5)} \\ u_5^{sc} &= \frac{y_5 r_5}{y_1 r_1 + y_5 r_5} & v_5^{sc} &= \frac{y_5 r_5}{\max(y_1 r_1, y_5 r_5)} \end{aligned}$$

Key parameters of the model are summarized in Table 13. More details about the model can be found in [79]. The overall structure of metabolic pathways and the model equations in [79] is similar to that presented here. The difference is that they refined the model according to the metabolic flux analysis of the experimental system to make the results physically meaningful. For example, the reaction $S_1 \rightarrow M_1$ is composed of two paths in the new model [79]; some reactions follow the substrate inhibition kinetics instead of simple Monod-type kinetics and the parameters have been optimized to closely model the experimental behavior. However, the overall structure of the model remains similar to the one considered in this work. The model used in this work captures the experimentally

Table 13: Rate constants and model parameters for the hybridoma system

r_1^{max}	r_2^{max}	r_3^{max}	r_4^{max}	r_5^{max}	r_6^{max}	r_7^{max}
0.05	0.032	0.05	0.003	0.01	0.05	0.032
K_1	K_2	K_3	K_4	K_5	K_6	K_7
0.004	0.0003	0.003	0.01	0.0001	0.05	0.0003
$r_i^{m,max} = 0.032$ and $K_{im} = 0.007$ for $i = 1, 2, 3, 7$						
$\alpha_i = 0.001$ for all i						
$\beta_i = \begin{cases} 0.05 & \text{if } i = 5, 6 \\ 0.001 & \text{otherwise} \end{cases}$						
$r_{ei}^* = 10^{-7}$ for all i						
K_{e1}	K_{e2}	K_{e3}	K_{e4}	K_{e5}	K_{e6}	K_{e7}
0.0001	0.0001	0.0001	0.00001	0.000001	0.00001	0.0001
y_1	y_2	y_3	y_4	y_5	y_6	y_7
1	0.7	0.5	1.65	0.6	1	0.72
$r_g^{'max}$	$K_g^{'}$		$y_{m_1 g}$	$y_{m_2 g}$	$y_{m_3 g}$	$y_{m_4 g}$
0.0575	0.01		0.4	0.1	0.1	0.4
			K_{g1}	K_{g2}	K_{g3}	K_{g4}
			5×10^{-7}	5×10^{-7}	0.005	0.005

observed steady state multiplicity, although it may not be as accurate quantitatively as the newest model of [79].

5.3 Successive Linearization-based MPC

The cybernetic models are highly nonlinear due to the Monod-type growth kinetics, dependence of metabolic rates on enzyme levels and cybernetic regulation variables. As a result, MPC scheme based on a single linear model could not be used for control of steady state switching in the reactor. A full nonlinear MPC was computationally infeasible to solve in a reasonable period of time because the model equations form a set of stiff ODEs. Hence, an MPC scheme based on successive linearization (represented as slMPC) of the model along the state trajectory in the prediction horizon was used [65]. The resulting prediction

equation was linear and the control move was computed by solving a Quadratic Program (QP), with the hessian and the gradient computed at each sample time based on the new linear approximation.

5.3.1 Model prediction

Let the governing model equations be represented as $\dot{x} = f(x, u)$. The discrete form of the model equations can be obtained by integrating the equations for one time step:

$$x_{k+1} = f_h(x_k, u_k) \triangleq \int_{h\cdot k}^{h\cdot(k+1)} f(x, u)\mathrm{d}t \tag{42}$$

One-step ahead prediction: In the model prediction step, the current state information is used to compute the information about the model trajectory in the future. The one-step prediction is given by

$$x_{k+1|k} = f_h(x_{k|k}, u_k) \tag{43}$$

At this stage, the control action u_k to be implemented is not known. In order to avoid solving a nonlinear optimization problem in determining this control action, a the model is linearized at the input value u_{k-1} to yield

$$x_{k+1|k} = f_h(x_{k|k}, u_{k-1}) + \mathcal{B}_{k|k}(u_k - u_{k-1}) \tag{44}$$

where, the continuous-time model is linearized

$$\mathcal{A}^c_{k|k} = \left.\frac{\partial f(x,u)}{\partial x}\right|_{x_{k|k},u_{k-1}} \qquad \mathcal{B}^c_{k|k} = \left.\frac{\partial f(x,u)}{\partial u}\right|_{x_{k|k},u_{k-1}}$$

and the resulting model discretized as $(\mathcal{A}_{k|k}, \mathcal{B}_{k|k}) = \texttt{c2d}^1(\mathcal{A}^c_{k|k}, \mathcal{B}^c_{k|k})$.

Multi-step Prediction: Following the same idea as above, we can write

$$x_{k+2|k} = f_h(x_{k+1|k}, u_{k-1}) + \mathcal{B}_{k+1|k}(u_{k+1} - u_{k-1}) \tag{45}$$

We linearize the above equation further with respect to $x_{k+1|k}$:

$$\begin{aligned} x_{k+2|k} \;=\; & f_h\Big(f_h(x_{k|k}, u_{k-1}), u_{k-1}\Big) + \\ & \mathcal{A}_{k+1|k}\Big[x_{k+1|k} - f_h(x_{k|k}, u_{k-1})\Big] + \mathcal{B}_{k+1|k}(u_{k+1} - u_{k-1}) \end{aligned} \tag{46}$$

[1]This MATLAB function converts continuous models to discrete form

Using Eq. (44), the term in square brackets above is just $\mathcal{B}_{k|k}(u_k - u_{k-1})$. Thus, we can write the two-step prediction as

$$x_{k+2|k} = f_{2h}(x_{k|k}, u_{k-1}) + \begin{bmatrix} \mathcal{A}_{k+1|k}\mathcal{B}_{k|k} & \mathcal{B}_{k+1|k} \end{bmatrix} \begin{bmatrix} u_k - u_{k-1} \\ \\ u_{k+1} - u_{k-1} \end{bmatrix} \tag{47}$$

Noting that

$$u_{k+1} - u_{k-1} = (u_{k+1} - u_k) + (u_k - u_{k-1}) = \Delta u_{k+1} + \Delta u_k,$$

we get

$$x_{k+2|k} = f_{2h}(x_{k|k}, u_{k-1}) + \begin{bmatrix} \mathcal{A}_{k+1|k}\mathcal{B}_{k|k} + \mathcal{B}_{k+1|k} & \mathcal{B}_{k+1|k} \end{bmatrix} \begin{bmatrix} \Delta u_k \\ \\ \Delta u_{k+1} \end{bmatrix} \tag{48}$$

We next make a simplification that the linearized matrices $\mathcal{A}_{k+i|k}$ and $\mathcal{B}_{k+i|k}$ are kept constant at their initial values $\mathcal{A}_{k|k}$ and $\mathcal{B}_{k|k}$. We will also drop the part $\cdot|k$ for notational convenience. Thus,

$$x_{k+2|k} = f_{2h}(x_{k|k}, u_{k-1}) + \begin{bmatrix} \mathcal{A}_k\mathcal{B}_k + \mathcal{B}_k & \mathcal{B}_k \end{bmatrix} \begin{bmatrix} \Delta u_k \\ \\ \Delta u_{k+1} \end{bmatrix} \tag{49}$$

Continuing further, we can write the multi-step prediction as:

$$x_{k+i|k} = f_{i\cdot h}(x_{k|k}, u_{k-1}) + \begin{bmatrix} \sum_{j=0}^{i-1} \mathcal{A}_k^j \mathcal{B}_k & \cdots & \mathcal{B}_k \end{bmatrix} \begin{bmatrix} \Delta u_k \\ \\ \vdots \\ \\ \Delta u_{k+i-1} \end{bmatrix} \tag{50}$$

5.3.2 Control implementation

A sequence of optimal moves that minimize the expected future error are computed based on the prediction equation. The controlled variable is a linear function of the system state: $y_k = Cx_k$. Hence, the desired prediction equation can be written as

$$\mathcal{Y}_k = \mathcal{S}^x + \mathcal{S}^u \Delta\mathcal{U} \tag{51}$$

where $\mathcal{Y}_k = [y_{k+1|k}^T, \cdots, y_{k+p|k}^T]^T$ is the vector of future predictions of the controlled variable, $\Delta\mathcal{U} = [\Delta u_k^T, \cdots, \Delta u_{k+m-1}^T]^T$ is the vector of future control moves, and the sensitivity matrices are given by

$$\mathcal{S}^x = \begin{bmatrix} Cf_h(x_k, u_{k-1}) \\ Cf_{2h}(x_k, u_{k-1}) \\ \vdots \\ Cf_{p.h}(x_k, u_{k-1}) \end{bmatrix}$$

$$\mathcal{S}^u = \begin{bmatrix} CB_k & 0 & \cdots & 0 \\ C(A_kB_k + B_k) & CB_k & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ C\sum_{i=0}^{p-1} A_k^i B_k & \cdots & \cdots & C\sum_{i=0}^{p-m} A_k^i B_k \end{bmatrix}$$

The following quadratic minimization is solved

$$\min_{\Delta\mathcal{U}_k} \|\Lambda^Q[\mathcal{Y}_k - \mathcal{R}]\|_2^2 + \|\Lambda^R \Delta\mathcal{U}_k\|_2^2 \tag{52}$$

to obtain the desired set of control moves within the horizon. Only the current control move Δu_k is implemented and the procedure is repeated at each sample time in a receding horizon fashion.

The steps in implementing the slMPC algorithm are summarized below.

- At each sample time, linearize the model equations described in sections 5.2.1 or 5.2.2 using the state estimate $x(k|k)$ and the previous control move u_{k-1} to obtain $\mathcal{A}_k$ and $\mathcal{B}_k$.

- Obtain the model predictions under constant input conditions within the prediction horizon by integrating the governing model equations to obtain the sensitivity matrix $\mathcal{S}^x$.

- Using the model predictions and the linear approximation, formulate and solve the optimization problem as a QP (quadratic program).

- Implement the current control move u_k and discard the other control moves computed. Repeat the process at each sample time.

- At the next time step, an observer such as extended Kalman filter may be used to obtain a state estimate based on fresh measurements. Using this estimate, the procedure is repeated all over again in a receding horizon manner. For a deterministic case, state estimation is not required because $x(k|k) = x(k)$.

5.4 Preliminary Results

Preliminary results of applying the slMPC algorithm for control of the microbial and the mammalian reactor are presented in this section. Two different cases of

- disturbance rejection in presence of step disturbances in the nutrient feed concentration, and

- optimal switching to drive the reactor to the desired steady state

are evaluated. The problem of steady state switching is especially challenging because of the non-differentiability of cybernetic regulation variable v associated with a metabolic shift.

5.4.1 Microbial reactor

As seen in Fig. 40, the working region in the microbial reactor is close to the turning point bifurcation. As a result, step disturbances in s_{2f} can drive the uncontrolled reactor to the undesirable steady state. We therefore consider the problem of disturbance rejection in presence of series of step changes in s_{2f}. Another interesting and more challenging problem is one of steady state switching. The problem of switching of steady states also assumes significance during start up of the reactor. Typically steady state switching is a difficult problem due to hard nonlinearity associated with the switch. The control objective is, therefore, to drive the reactor from the low biomass steady state to the desirable high biomass yield state. It may be viewed as a step change in the setpoint at time $t = 0$ from

the low biomass to the high biomass yield steady state. The performance of the controller is evaluated under step disturbances of various magnitude in parameter s_{2f}.

5.4.1.1 Disturbance rejection

First, we consider the problem of disturbance rejection in the microbial reactor. Disturbance enters the system in the form of step variations in the feed s_{2f} concentration. We consider a specific case of a series of step changes that result in a catastrophic shift in the steady state, causing the reactor to go to the other steady state if left uncontrolled. Figure 42 shows the input step disturbance to the system and its catastrophic effect on the uncontrolled reactor.

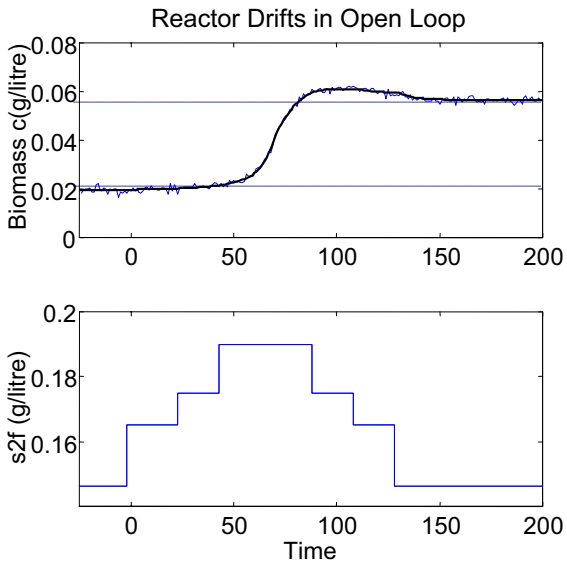

Figure 42: Step disturbances in feed s_{2f} concentration. If left uncontrolled, the reactor drifts to the other steady state.

After performing a number of simulation studies, we found that slMPC was able to give a satisfactory control performance for the system, and that a full nonlinear MPC was not required. In all the simulations, a measurement noise was added and changes in s_{2f} were assumed to be known or measured. All the results represent a control horizon of

$m = 10$ and prediction horizon of $p = 40$. Figure 43 shows the control result when only the cell concentration was controlled using both dilution rate D and glucose feed rate s_{1f} as manipulated variables. The output penalty weight used was $\lambda^y = 1$ and penalty for input rate was $\lambda^u = \text{diag}[2.5, 0.5]$. This results in an interesting scenario. The cell concentration is kept at the steady state. However, the cells are in a different metabolic state, as indicated by other output variables (s_1 and s_2) and the cybernetic regulation variables v_1 and v_2. Thus, the reactor displays input multiplicity in addition to the output multiplicity and the desired control goal is not achieved.

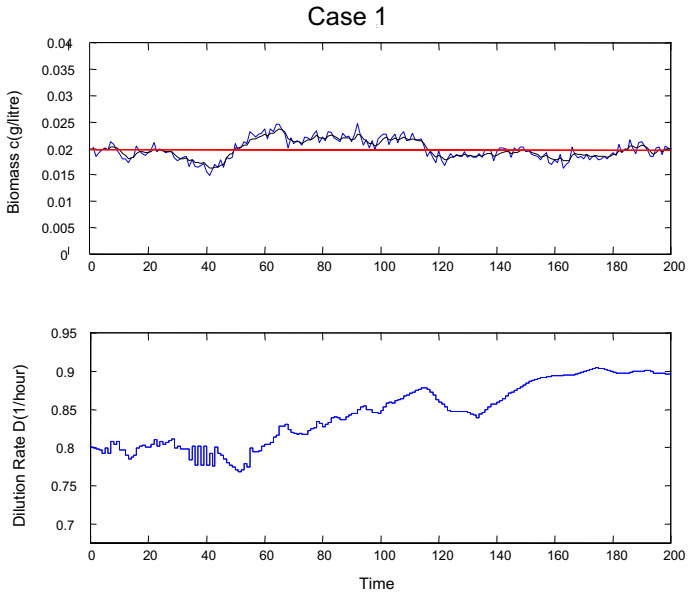

Figure 43: Results of controlling cell concentration using dilution rate and inlet s_{1f} concentration. Although the cell concentration reaches its desired value, the system is in a physiologically different steady state.

Next, we considered a multi-output single-input control strategy. In this case, effluent s_2 concentration was also controlled in addition to the cell mass. Only the dilution rate was used as a manipulated variable. In this situation, as indicated in Figure 44, the system was maintained at the appropriate steady state.

The results of applying slMPC for the practically relevant problem of controlling the

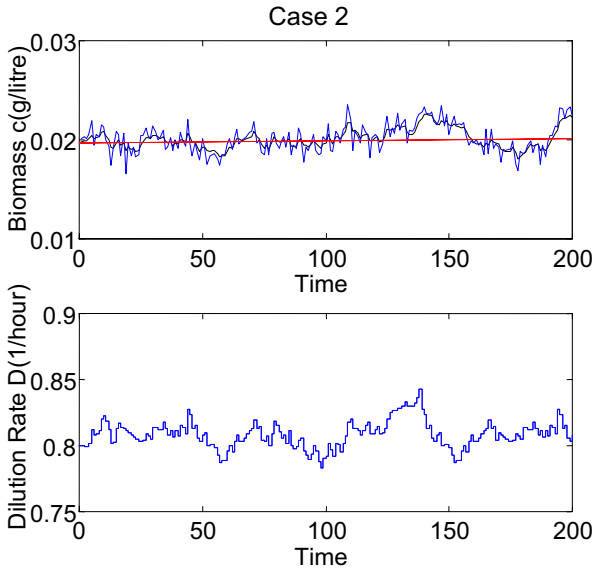

Figure 44: Same as Figure 43, except that effluent s_2 concentration is also measured in addition to the biomass. In this case, the system is controlled at the desired steady state

system at the desired high biomass steady state are skipped. This steady state does not display input multiplicity, and the control results are similar to those observed in case 2 discussed here. A more interesting problem, that of optimal switching from the low biomass to the high biomass yield state, is considered next.

5.4.1.2 Steady state switching

Figure 40 shows the proximity of the working steady state to the turning point bifurcation. As a result, the reactor could reach the undesirable steady state either due to the effect of disturbances or during reactor startup. The aim is therefore to optimally drive the reactor to the desired steady state. One can view this as a step change in the setpoint. A deterministic problem will be considered, with known / measured variations in s_{2f} acting as disturbance to the system. Again, for this problem, we considered $m = 10$, $p = 40$, $\lambda^y = 1$ and $\lambda^u = 2.5$. The results of applying slMPC for steady state switching problem are shown in Figure 52 as a thick solid line.

The main issue in application of slMPC for this system lie in the large computational burden and the sluggish controller behavior. Computational burden is large because a nonlinear model has to be employed to make predictions within the horizon for computing the optimal control moves online. Additionally, the long-ranged transients of the system demand larger horizon sizes. While a controller can be more aggressively tuned (lower λ^u) for faster steady state switching, this affects the disturbance rejection performance (previous subsection) adversely.

5.4.2 Steady state switching in hybridoma reactor

After considering the steady state switching in a microbial reaction, we turn our attention to a more difficult problem of a mammalian reactor. Compared to the microbes, mammalian cells have a more sluggish response, greater nonlinearity, lower tolerance to external factors (temperature, pH etc) and higher sensitivity to abrupt changes in the environment. Hence, control of mammalian cultures is significantly tougher than a microbial reactor. The cybernetic model, discussed in section 5.2.2, reproduces the experimentally observed three distinct steady states. We label these steady states as "low," "intermediate" and "high" biomass yield steady states. From the point of view of the model, a hybridoma system is tougher to control and analyze than the microbial system because it consists of fifteen states, four local cybernetic competitions, one global cybernetic competition and slower dynamics.

First, we consider driving the reactor from the low biomass steady state to the high biomass steady state. Figure 45 shows the response of the model to a step change in dilution D at the low biomass steady state. The long-range transient response of the system is evident from this figure: the output trajectories are nearly flat for a long time, until a catastrophic shift is encountered around $t = 460$, which causes a change in the metabolic state and a sharp change in the measurable outputs, viable cell concentration and effluent substrate concentration(s). Note also the non-differential nature of the cybernetic variable v_7, which abruptly takes a value of $v_7 = 1$ (fully activated enzyme state) and triggers a catastrophic event.

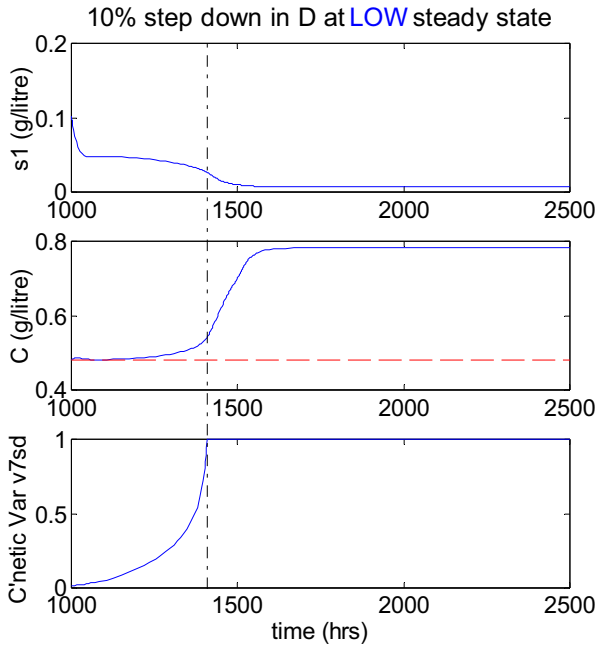

Figure 45: Response of the hybridoma reactor to 10% step change in dilution rate at the low biomass yield steady state

Figure 46 shows the results on applying slMPC for switching of the hybridoma reactor from the low biomass steady state to intermediate and then to the high biomass yield steady state. The dilution rate was used as the manipulated variable, while the viable cell concentration and glucose concentration were used as control variables. In this case, we chose $m = 15$, $p = 50$, $\lambda^y = \text{diag}[1, 0.1]$, $\lambda^u = 0.2$ and appropriate constraints for the dilution rate ($0.01 \leq D \leq 0.05$, and $|\Delta D| \leq 0.0033$). The "oscillatory" input response can be perhaps avoided by increasing the penalty on input moves, but that makes the controller very sluggish.

Next, we used the same tuning parameters for steady state switching from high biomass yield steady state to the intermediate biomass yield steady state. Figure 47 shows that the controller is unable to drive the reactor to the desired steady state. The manipulated variable saturates at the constraint ($D = 0.01$). The reason for this can be seen in Fig. 48, which shows that the system exhibits long-range transient inverse response to a step change in dilution rate. With an increase in dilution rate, the viable cell concentration increases

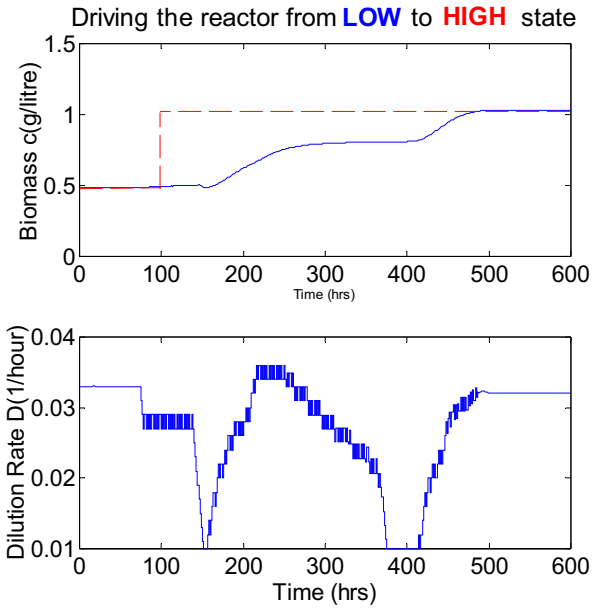

Figure 46: Driving the hybridoma reactor from low biomass steady state to the high biomass steady state.

until a certain point. Then, there is a sharp drop in the cell concentration due to metabolic shift. As a result, the controller decreases the dilution rate until it saturates at the lower constraint. No change in the metabolic state is therefore observed. In order to accurately track the system behavior, a very long prediction horizon will be needed.

An alternate way is to realize that the response of s_1 to the step change in dilution rate is relatively monotonic. In other words, as seen in the top plot of Figure 48, a step up change in dilution rate increases the effluent s_1 concentration gradually until the metabolic shift takes place causing a large increase in s_1. Hence, at the high biomass yield state, we decided to penalize only the substrate concentration s_1. Thus, if the system is at high biomass steady state, we use $\lambda^y = \text{diag}[0, 1]$. When a metabolic shift is detected though the cybernetic variable v_1, we change the penalty terms back to $\lambda^y = \text{diag}[1, 0.1]$. The result of applying this strategy is shown in Figure 49. Clearly, with this strategy, the controller drove the system to the intermediate steady state.

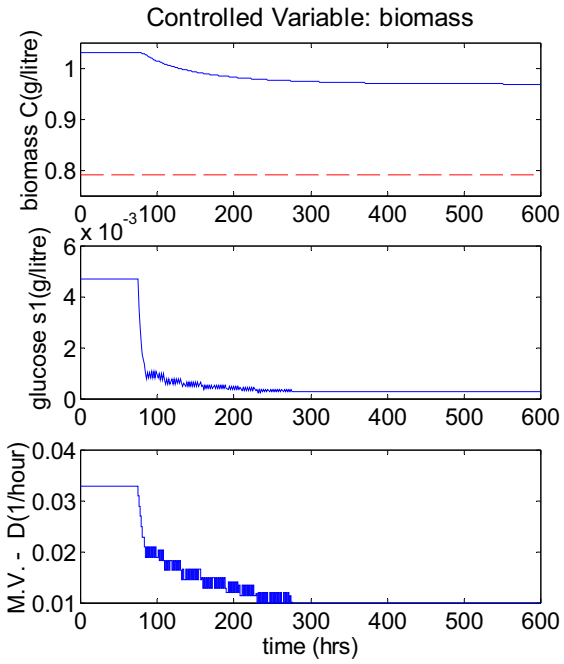

Figure 47: The controller is unable to drive the reactor switching from the high to the intermediate biomass yield steady state.

5.5 Conclusions

The control of a microbial bioreactor and a mammalian bioreactor, both displaying steady state multiplicity was considered in this chapter. The reactors were modeled using the cybernetic modeling framework and sequential linearization-based Model Predictive Controller (slMPC) was applied for this system. Although slMPC shows a reasonable control performance, the nonlinearity and the long range transient response of the system results in large computational load. In the subsequent chapters, we will address the issue of optimality and online computation for optimal steady state switching in the microbial reactor for a deterministic case of step disturbances in s_{2f}.

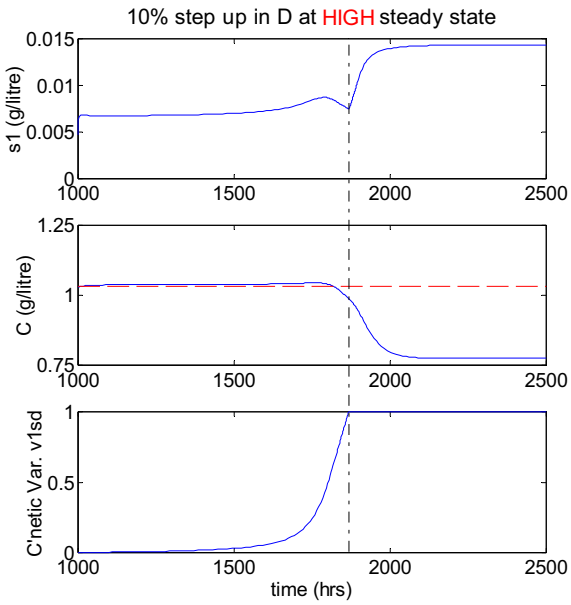

Figure 48: Open loop response of the hybridoma reactor to a 20% step up change in dilution rate at the high biomass yield steady state.

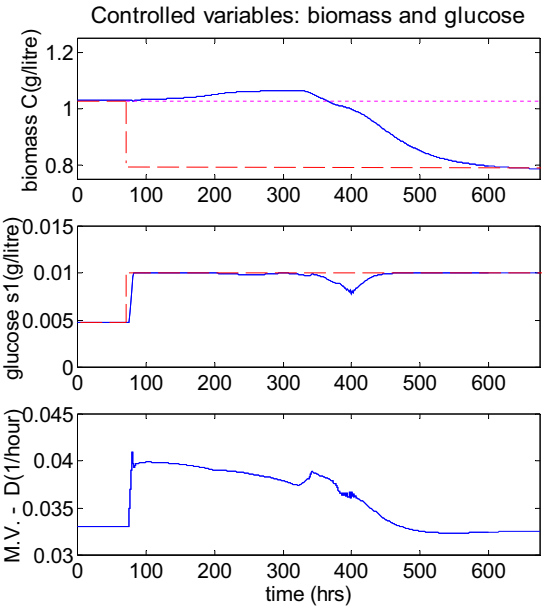

Figure 49: Switching from the high biomass steady state to the intermediate biomass steady state is achieved using modified penalty weights for the controlled variables.

CHAPTER VI

SIMULATION-BASED METHOD FOR OPTIMAL STEADY STATE SWITCHING IN A BIOREACTOR

The problem of driving a bioreactor to the desired steady state using slMPC was presented in the previous chapter. For the culture of bacterial cells, the controller performance was sluggish and the online computational time was significantly large. To alleviate these problems, a simulation-based approximate Dynamic Programming (simDP) method is introduced in this chapter and applied to the *Klebsiella oxytoca* example.

6.1 Introduction

In optimal control, one is often faced with the task of solving nonlinear optimization problems — either off-line or more commonly on-line. An example is the popular method of Model Predictive Control (MPC) [78, 76, 63], which requires a nonlinear dynamic optimization problem cast over a prediction window to be solved at each sample time when applied on a nonlinear process model. Such problems also arise in batch processes where operating recipes that minimize the batch time or maximize the product quantity and quality are desired. Optimization problems involving nonlinear dynamic models are intrinsically hard problems and it is difficult to assure the attainment of quick, reliable solutions, which are needed in most of the practical applications. Difficulties exist even in off-line optimization, when the problem involves a model of high dimension and a large time window, thus yielding a large set of optimization variables and constraints. In practice, these problems are often solved in a highly approximate sense (by using a linear approximation of the model, for example) or are avoided by adopting heuristic policies instead.

One approach for solving dynamic optimization is Dynamic Programming (DP). Here, the aim is to find the optimal *'cost-to-go'* function, which can be used to parameterize the solution with respect to system state – either as a continuous function or as a lookup

table – thereby simplifying the task of obtaining on-line solutions. However, the approach is largely considered impractical as analytical solution of resulting dynamic program is seldom possible and numerical solution suffers from the 'curse of dimensionality' [8].

Neuro-Dynamic Programming (NDP) was proposed as a way to alleviate the curse of dimensionality [12]. It uses simulated process data obtained under suboptimal policies to fit an approximate cost-to-go function – usually by fitting artificial neural networks, hence the name. In value iteration approach of NDP, the initial approximate cost-to-go function is further improved by an iteration procedure based on the so called Bellman equation. In this context, the simulation's role is two-fold. First, by simulating the process under a reasonably chosen suboptimal policy and all possible operating parameters / disturbances, it provides a set of data points that define the relevant or "working" region in the state space. Second, the simulation provides the cost-to-go value under the suboptimal policy for each state visited, with which iteration of the Bellman equation can be initiated. This method is closely related to the efforts of Artificial Intelligence community in Reinforcement Learning (RL), where an agent learns by interacting with the environment and recording a fitness value as a function of the system state. The NDP / RL approach has received significant attention for its successes in several applications such as elevator dispatch problem and a program that plays Backgammon at the world championship level. Extension of these approximate DP concepts to the control of process systems is not a trivial task. Recently, the key issues involved in implementing the simulation-based approximate DP method were addressed by a colleague Jong Min Lee in his PhD thesis [66].

In this chapter, the simulation-based DP approach is applied to optimal control of steady state switching in a continuous bioreactor. We consider the growth of bacterial cells (such as *Klebsiella oxytoca*) on multiple substitutable nutrients. The cybernetic modeling framework developed by Ramkrishna and coworkers (see [62], [111] and references therein) is used to model the system. This non-linear system is characterized by multiple steady states and long-lived transient behavior, which demands that a nonlinear optimization cast over a large time window be solved at each sample time. In the previous chapter, we had applied a MPC method based on successive linearization of the nonlinear model [65] to this

system. Long prediction and control horizons were shown to be necessary to control the reactor at desired steady state because of certain peculiar dynamics of the bioreactor, *e.g.*, quickly settling to an almost stable behavior that lasts for a long period followed by a sharp drift to another steady state, which is "triggered" by a change in the cells' metabolic states.

We seek to use this approach not only to reduce the on-line computational demand but also to improve the controller performance, through the use of cost-to-go approximator. A neural network is chosen as an approximator to obtain cost-to-go as a function of system states. While a properly trained neural network has good interpolation capabilities, one may not use it to extrapolate over regions of state space not covered during its training. Extrapolation by neural network is shown to result in deteriorated performance of the controller. In turn, we discuss possible remedies, either by confining the control calculations to the visited regions of the state space or by expanding coverage through additional simulations.

6.2 Mathematical Preliminaries

Optimal control involves finding control actions that minimize certain multi-stage objective function for a horizon of size p. Mathematically, the resulting general dynamic optimization problem for a given initial state x_0 may be represented as follows:

$$\min_{u_0,\ldots,u_{p-1}} \sum_{i=0}^{p-1} \phi(x_i, u_i) + \bar{\phi}(x_p) \tag{53}$$

subject to

$$\begin{aligned} \text{Path Constraint:} \quad & g_i(x_i, u_i) \geq 0, 0 \leq i \leq p-1 \\ \text{Terminal Constraint:} \quad & \bar{g}(x_p) \geq 0 \\ \text{Model Constraint:} \quad & x_{i+1} = f_h(x_i, u_i) \end{aligned}$$

In the above, x_i represents the state at the i^{th} sample time, $\phi(x, u)$ is the single stage cost function and $\bar{\phi}$ is the terminal cost. The path/terminal constrains as well as the system model may be nonlinear. A continuous model $\dot{x} = f(x, u)$ can be discretized by integrating it over the sampling time; *i.e.*, $f_h(x_i, u_i) \equiv \int_h f(x_i, u_\tau) d\tau$, with piece-wise constant input $u(\tau) = u_i$ for $i \cdot h \leq \tau < (i+1) \cdot h$.

The above problem (53) is a general representation of a stagewise dynamic optimization problem. It may be solved offline to obtain an open loop optimal target trajectory as in batch or semi-batch systems. It can also be used to generate a state feedback policy, which relates a system state to optimal actions. It can also be solved at the end of an episode in run-to-run or period-to-period control. This problem is also encountered in planning and scheduling, stochastic shortest path problems, agent-based learning, etc. The focus of this part of the thesis is on online optimal control of discrete-time systems with continuous state and action (input) space.

6.2.1 Receding horizon control

For a continuous system, the problem (53) above is often reinitialized using the feedback update of the current state and solved online at each sample time in order to find the optimal input adjustment for the given state — as in 'Receding Horizon Control' [78, 76, 63]. The slMPC algorithm is an example of such a method, which employs quadratic performance criteria and linear approximations of the model within the prediction window.

Often, it is shown to be advantageous to solve an infinite horizon problem (in which p is set to infinity). The ∞-horizon equivalent of (53) can be expressed as

$$\min_{u_0,\ldots} \sum_{i=0}^{\infty} \phi(x_i, u_i) \tag{54}$$

The same set of model and path constraints should be enforced here. Terminal constraints and penalty function may be imposed to convert the above infinite horizon problem into an equivalent but numerically tractable finite horizon problem. However, the computational load can be significant, especially when the window size is large and nonlinear dynamics are involved. Often, its impossible to solve a nonlinear optimization problem; hence suboptimal strategies such as linearization (eg. slMPC) or heuristic control schemes are employed. Finally, the receding horizon control is an open loop optimal technique with frequent feedback correction; it is not closed loop optimal.

6.2.2 Dynamic Programming

Dynamic Programming (DP) is an elegant way to solve the previously introduced (closed loop) optimization problem. It is based on *Bellman's optimality principle*, which states that if a control policy $u_i = \mu_i^*(x_i)$ $(i = 0, \ldots, p-1)$ is the optimal policy for the original problem (53), then the truncated policy $\{\mu_k^*, \ldots, \mu_{p-1}^*\}$ is also optimal for the subproblem:

$$\min_{u_k,\ldots,u_{p-1}} \sum_{i=k}^{p-1} \phi(x_i, u_i) + \bar{\phi}(x_p) \tag{55}$$

irrespective of how we reach a state x_k at time k.

Intuitively, this means that for a path going from a point A to another point B, passing through a point X at some stage k to be optimal, the path from X to B also has to be optimal. This is true irrespective of how one reaches X. If it isn't, then the cost can be reduced by taking the more optimal path from X to B. The theoretical proof of the above statement can be found in any standard DP text [10, 87].

At this point, we formally introduce the concept of "policy" as a rule by which control actions are chosen. A policy is in general dependent on the system state. Thus $u_k = \mu(x_k)$ represents the control action taken at time k according to the policy $\mu(x)$, where μ maps the states x_i to the control actions u_i.

6.2.2.1 Cost-to-go function

Taking control actions based on any policy μ is associated with certain cost according to the performance function (53). The total cost incurred from time k to the end of the horizon is called *cost-to-go*. The performance function in Eq. (53) consists of two parts: the current cost $\phi(x_k, u_k)$ incurred at the current state x_k on implementing control action u_k and the sum of future costs incurred until the end of horizon. The latter is expressed as *value* or *cost-to-go* function, which expresses the desirability of state x_{k+1}.

For Eq. (53), the cost-to-go at each stage is defined as

$$J_i = \min_{u_{p-i},\ldots,u_{p-1}} \left\{ \sum_{j=p-i}^{p-1} \phi(x_j, u_j) + \bar{\phi}(x_p) \right\} \tag{56}$$

Then, the calculation of the cost-to-go function at each stage can be done recursively as

$$J_i(x) = \min_u \left\{ \phi(x, u) + J_{i-1}(f_h(x, u)) \right\}, \tag{57}$$

where $f_h(x,u)$ denotes the successor state generated by the state transition model. The above is sequentially solved from $i=1$ through $i=p$ with the initialization of $J_0=\bar{\phi}(x)$, with the pertinent terminal / path constraints imposed at each stage.

For the infinite horizon problem of (54), cost-to-go can be similarly calculated as

$$J_\infty(x)=\min_u\{\phi(x,u)+J_\infty(f_h(x,u))\} \tag{58}$$

6.2.2.2 The objective of DP

The objective of DP is to calculate the optimal cost-to-go function $J^*(x)\ \forall x\in\mathcal{X}$, the $*$ representing optimality. The optimal cost-to-go function can be computed offline, either analytically (as in Linear Quadratic Regulator, LQR) or numerically. Optimal cost-to-go can be shown to satisfy some form of the Bellman Equation [10]

$$J^*(x)=\min_u\{\phi(x,u)+J^*(f_h(x,u))\} \tag{59}$$

The cost-to-go function, once obtained, represents convenient means to compute the optimal control action for any general state x_0 by solving the following equivalent one-stage problem:

$$\mu^*(x_0)=\arg\min_{u_0}\{\phi(x_0,u_0)+J^*(f_h(x_0,u_0))\} \tag{60}$$

In general, the DP solution would represent a much smaller online computational load than the receding horizon control approach by characterizing the optimal solution of (53) or (54) as a cost-to-go function and converting the multi-stage problem (53) to an equivalent single-stage problem (60).

6.2.3 Conventional DP algorithms

In this section, two popular algorithms used in DP to obtain the optimal cost-to-go function are introduced; viz. value iteration and policy iteration. Some key properties of the DP operator are first defined. Let T be the infinite-horizon α-discounted DP operator for a system

$$x_{k+1}=f_h(x_k,u_k)$$

defined as:

$$(TJ)(x_k) = \min_{u_k} \{\phi(x_k, u_k) + \alpha J(f_h(x_k, u_k))\} \tag{61}$$

The DP operator has following properties [10, 12]

- MONOTONICITY: If $J \geq \bar{J}$, then $TJ \geq T\bar{J}$
- CONVERGENCE: The sequence $\{T^N J\}$ converges for any J
- OPTIMALITY: $J^* = \lim_{N \to \infty} T^N J$

Lee [66] provides an excellent review of conventional DP methods and issues related to the implementation of approximate DP. The proofs of above properties can be found elsewhere [10, 66]. The implications of the above properties in conventional DP are that DP operator is guaranteed to improve the cost-to-go function and that the DP sequence approaches the optimal cost-to-go asymptotically. The algorithms given below can therefore be used to obtain evolutionary improvement in cost-to-go.

6.2.3.1 Value Iteration

Value iteration, as the name suggests, iterates on the cost-to-go or value function for each state until convergence. In other words, it sequentially improves the cost-to-go function starting from a suboptimal cost-to-go J^0 using the Bellman equation (59) as an update rule. For any state x, value iteration can be written as:

$$J^{i+1}(x) = \min_{u} \left\{\phi(x, u) + J^i\left(f_h(x, u)\right)\right\} \tag{62}$$

The cost-to-go function J^{i+1} is an improvement over J^i, except when J^i is optimal, in which case both are equal [12]. Value iteration thus generates a sequence of monotonically improving cost approximations [104]:

$$J^0(x) \xrightarrow{\min} J^1(x) \xrightarrow{\min} \cdots J^i(x) \xrightarrow{\min} \cdots \xrightarrow{\min} J^*(x)$$

6.2.3.2 Policy Iteration

Policy iteration is a two step procedure that involves finding an improving policy $\mu^{i+1}(x)$ using cost-to-go function $J^i(x)$, followed by updating the cost-to-go function $J^{i+1}(x)$ based

on this policy. We thus obtain a sequence of monotonically improving policies and cost approximations [104]:

$$\mu^0(x) \xrightarrow{J^\mu} J^0(x) \xrightarrow{\min} \mu^1(x) \xrightarrow{J^\mu} J^1(x) \xrightarrow{\min} \cdots \mu^*(x) \xrightarrow{J^\mu} J^*(x) \xrightarrow{\min}$$

The first step, known as *policy improvement*, involves solving a minimization problem to find an improved policy:

$$\mu^{i+1} = \arg\min_u \left\{ \phi(x,u) + J^i\left(f_h(x,u)\right) \right\} \tag{63}$$

The second step involves following the policy to compute cost-to-go function and is known as *policy evaluation*. Policy evaluation itself is an iterative step, in which starting from $J^{i+1,0} = J^i(x)$, we use

$$J^{i+1,j+1} = \phi(x, \mu^{i+1}(x)) \ + \ J^{i+1,j}\left(f_h(x, \mu^{i+1}(x))\right) \tag{64}$$

until convergence. The cost-to-go obtained on convergence of policy evaluation is represented as J^{i+1}.

Policy improvement and policy evaluation are performed sequentially until convergence. It can be proved that μ^{i+1} obtained according to (63) is better than policy μ^i. Indeed when the cost-to-go function J^i is optimal, the two are equal.

6.2.3.3 Temporal Difference-Based λ-Policy Iteration

Its well known that the value iteration converges asymptotically to the optimal cost-to-go function. On the other hand, policy iteration converges finitely and requires lesser iterations than value iteration [12]. However, policy evaluation required to obtain the cost-to-go function J^μ for the policy μ is an iterative process and it may converge very slowly[1]. This is especially true when the number of states is large. The λ-policy iteration method proposed by Bertsekas and Ioffe [11] aims at accelerating the policy evaluation step by introducing a "discount factor" $\lambda \in [0 \quad 1]$.

The underlying idea is that the discount factor does not alter the cost structure for a given policy if the one-stage cost for that policy is 0. The concept is based on Temporal

[1]This is a standard result in DP. See chapter 2 in [12] for more discussion

Differences (TD), which are at the heart of powerful algorithms called "TD learning" that were introduced by the RL community.

We will use TD associated with each state transition x_k to x_{k+1}:

$$d(x_k) \stackrel{\Delta}{=} \phi(x_k, \mu(x_k)) + J(x_{k+1}) - J(x_k) \tag{65}$$

to update the cost-to-go. Using this transformation, we can perform policy evaluation as:

$$J^{i+1}(x_k) = J^i(x_k) + \sum_{m=0}^{\infty} \lambda^m d(x_{k+m}) \tag{66}$$

where $d(x_k)$ and λ are as defined above. Note that when cost-to-go function converges, $d(x_k)$ as defined by (65) equals 0. Thus the overall cost structure is not altered. Clearly, this is a λ-weighted DP problem and is appropriately named λ-policy iteration [11] and not TD learning. Bertsekas and Ioffe [11, 12] prove convergence properties for this method and show that it reduces to value iteration for $\lambda = 0$ and policy iteration for $\lambda = 1$.

6.2.4 Simulation-based Approximate Dynamic Programming

In very few cases can we solve the stagewise optimization analytically to obtain a closed-form expression for the cost-to-go function. With notable exception of linear quadratic control (LQG) applicable for unconstrained linear systems, analytical solutions to Bellman equation (59) cannot be found. The numerical approach to the problem involves gridding the state space, calculating and storing the cost-to-go for each grid point as one marches backward from the last stage to the first using Eq. (57). Alternatively, conventional value or policy iteration techniques described in the previous section can be used iteratively to solve the optimization problem for each point in the state space. As the state dimension increases, the memory requirements for storage of the computed cost-to-go values as well as the computational requirements in solving the iterative optimization grow exponentially. Thus, this approach suffers from the famous *curse of dimensionality* due to this exponential increase in computational requirements with increasing state dimension.

Researchers in NDP and RL community have developed various model-based and model-free algorithms that avoid the curse of dimensionality associated with conventional DP. The common features of these algorithms are

- They aim to obtain an *approximation* of the optimal cost-to-go function.
- They start with a suboptimal policy and a suboptimal cost-to-go, using each to recursively improve the other.
- They do so through synergistic use of *simulations* and *function approximations*. The iterations are performed in a restricted region of the state space defined through simulations and the function approximator is used to interpolate the cost-to-go values within this restricted region.

Given a state transition model, a conceptually straightforward method is to use the conventional value or policy iteration algorithms. Exhaustive sampling of state space is avoided by identifying relevant regions of the space through simulation under judiciously chosen suboptimal policies. The role of suboptimal policies is two-fold: to provide an initial policy and cost-to-go approximation, and to define the visited region of the state space over which the iterations are performed. The algorithms are described in Appendix B.

In this chapter, simulation-based Dynamic Programming (simDP) method is applied to the steady state switching in the bioreactor considered in section 5.2.1. Value iteration algorithm is used to obtain the cost-to-go function. The cost-to-go function so obtained is then used for online control. We demonstrate the need to have adequate coverage of the state space and restrict the controller to the visited regions of the state space for good performance. In the next chapter, we further investigate the salient features of simulation-based value and policy iteration algorithms and point out the key issues involved with implementation of simDP.

6.3 Application of simDP to a Microbial Cell Reactor

Application of simDP method to optimal control of steady state switching in a continuous culture of *Klebsiella oxytoca* growing on a mixture of two nutrients glucose and arabinose is considered in this section. This system model was described in section 5.2 and control using successive linearization-based MPC (slMPC) was analyzed in section 5.4. In this section, the efficacy of simDP method over slMPC is presented.

6.3.1 Suboptimal control law: slMPC

The successive linearization based slMPC algorithm [65] was used as the initial suboptimal control law. This method linearizes the nonlinear model at each current state and input values to compute a linear prediction equation. The control moves are computed by solving a QP, with the hessian and the gradient computed at each sample time based on the new linear approximation. Detailed description of this method was provided in the previous chapter (section 5.3).

The closed loop simulation with the slMPC algorithm under a parameter conditions $s_{2f} = 0.146$ and $c_{SP} = 0.055$ is shown in Figures 50, 52 as thick line. At time $t = 0$, the system was at the low biomass steady state, when a step change in the set point to the high biomass steady state was applied. The constraints on the dilution rate were chosen to be $u_{\min} = 0.6$, $u_{\max} = 1.0$ and $\Delta u_{\max} = 0.05$, keeping in mind that the model is not valid for low dilution rates and to avoid washout condition that occurs at high dilution.

6.3.2 Obtaining optimal cost-to-go function approximator

6.3.2.1 Simulations using suboptimal controller

The suboptimal slMPC controller described above, was used to obtain closed loop simulation data for the proposed strategy. It was implemented for four values of $s_{2f} =$ [0.14 0.145 0.15 0.155], to cover the possible range of variations. For each of the parameter values, the reactor was started at three different $x(0)$ values around the low biomass yield steady state. We obtained 100 data points for each run. Thus a total of 1200 data points were obtained. The infinite horizon cost-to-go values were computed for all the 1200 points. Note that the calculated cost-to-go value is approximate infinite horizon cost, as described in Appendix B.1.

6.3.2.2 Cost approximation

States were augmented with the parameter s_{2f} (see Table 10). A functional approximation relating cost-to-go with augmented state was obtained by using neural network — a multi-layer perceptron with five hidden nodes, six input and one output node. The neural network

showed a good fit with mean square error of 10^{-3} after training for 1000 epochs. This is the zeroth iteration, denoted as $\tilde{J}^0(x)$.

6.3.2.3 Improvement through Bellman iterations

Improvement to the cost-to-go function is obtained through iterations of the Bellman equation (62). This method, known as value iteration (or value iteration), is described in Appendix B.1. The solution of the one-stage-ahead cost plus cost-to-go problem, results in improvements in the cost values. The improved costs were again fitted to a neural network, as described above, to obtain subsequent iterations $\tilde{J}^1(x)$, $\tilde{J}^2(x)$, and so on ..., until convergence. Cost was said to be "converged" if the sum of absolute error was less than 5% of the maximum cost. The cost converged in 4 iterations for our system.

6.3.3 Online implementation

The converged cost-to-go function from above was used online in solving the one stage ahead problem. The control move was calculated as in 67 and implemented online in a receding horizon manner.

$$u(k) = \arg\min_{u(k)} \left\{ \phi\left(x(k), u(k)\right) + \tilde{J}^4\left(f_h(x(k), u(k))\right) \right\} \tag{67}$$

The results are shown as broken line in Figure 50 and a numerical comparison is shown in Table 14. The method was tested for various s_{2f} values. Representative results for a single s_{2f} value of 0.146 are shown. First two rows in the table represent the online performance of the two approaches, viz. slMPC and the proposed simDP scheme. In the table, the last two columns show the comparison between the two schemes; the first four columns represent the control algorithm, the number of data points used in obtaining cost-to-go function, the number of cost-iterations and the number of hidden nodes in the neural network approximation of converged cost function.

Clearly, the new scheme performs worse than the original slMPC scheme. An overshoot is observed and the total cost is also increased. However, there is a dramatic reduction in computational time — from almost half an hour to under 2 minutes, for 100 time steps (50

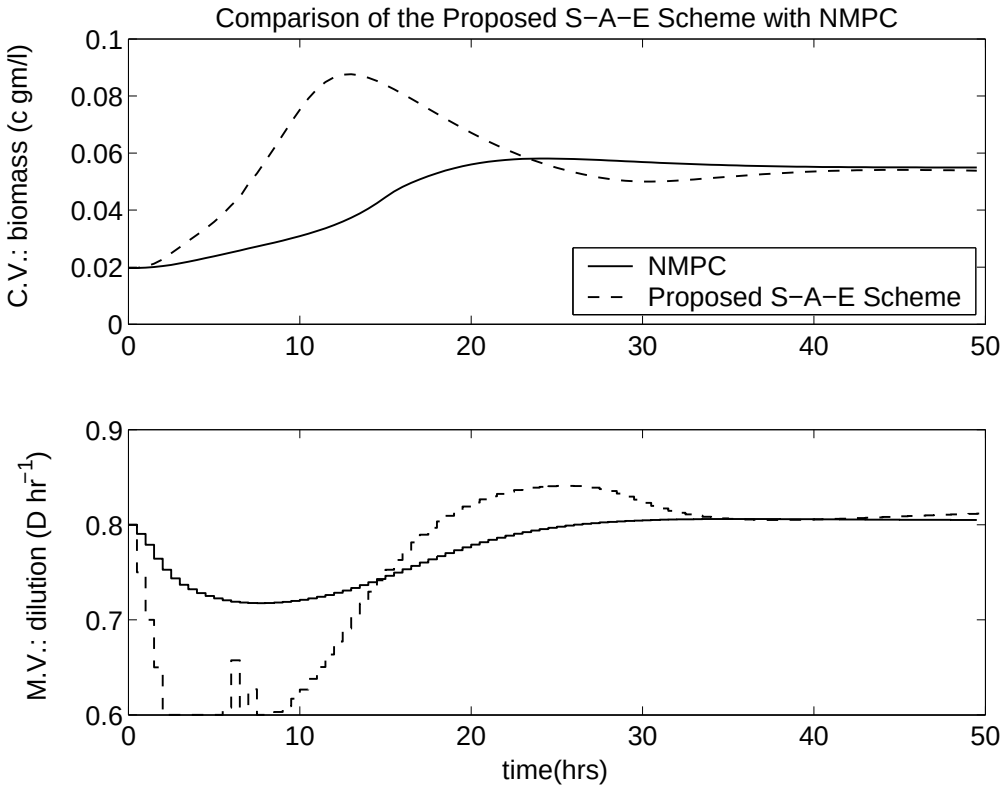

Figure 50: Comparison of the online performance of slMPC control law and simDP approach. The simDP controller gives faster response initially, but causes significant overshoot, as compared to more sluggish slMPC controller. Numeric comparison between the two approaches is shown in table 14.

hours). In the next section, we evaluate the possible reasons for the worse behavior and discuss possible solutions.

6.3.4 Improvement in the Strategy

The policy improvement theorem, described earlier, indicates that the use of converged solution from Bellman iterations is expected to improve the performance over the suboptimal controller. At worst, the performance of the proposed scheme should be at par with the original suboptimal scheme. The logical reasoning behind this is that Bellman iterations should choose the original policy over all other policies that lead to a less optimal result (for mathematical proof, please see [104] or references therein).

The possible causes of error could either be presence of local minima, poor fitting of the cost approximations (by the neural network), or extrapolation to previously unvisited

Table 14: Details of nonlinear MPC algorithm v/s new scheme and its modifications. †Additional simulations for increasing data coverage; ‡Intel Pentium III, 800 MHz processor, 512 MB RAM, running Matlab 6 Release 12 on Windows 2000.

Control Algorithm	Number of data points	Cost Iterations	Number of hidden nodes	Total cost (at $x(0)$)	CPU Time (seconds)‡
slMPC	- N.A. -	- N.A. -	- N.A. -	22.54	1080.3
New Scheme	1200	4	5	24.18	98.7
w/ Restricting	1200	2	4	9.06	127.7
w/ Add Sim†	2088	4	5	9.37	79.5
w/ Policy update	1395	7	9	10.32	74.12

regions in state space. An investigation of the state space plot in Figure 51 suggests that extrapolation to previously unvisited regions of the state space could have resulted in deterioration of the controller performance. The system spans 6-D space (5 states and 1 parameter s_{2f}). All the state space plots are projections of the 6-D space on a 2-D plot, with $[s_2]$ as the abscissa and biomass $[c]$ as the ordinate. Possible remedies, which are discussed in this section, could be to restrict the control calculations to the visited part of state space or include additional data to increase the coverage as necessary.

6.3.4.1 Gridding and restricting the working region

In this method, the optimizer was *restricted* to search only in the visited region of the state space during both offline Bellman iterations as well as online implementation. The state space was grid into 10 grid points per state (10^5 cells) and each cell was identified as "visited" or "unvisited". A cell was said to be "visited" if it contained at least one data point. The neural network was used for the visited cells, whereas unvisited cells were associated with a very high cost, thus creating an artificial boundary to restrict the search within visited cells.

In this case, convergence of the Bellman equation occurred in 2 iterations. The neural network approximation required 4 hidden nodes. The results are shown in Figure 52 and numeric values in Table 14. The method is a clear improvement over the slMPC control

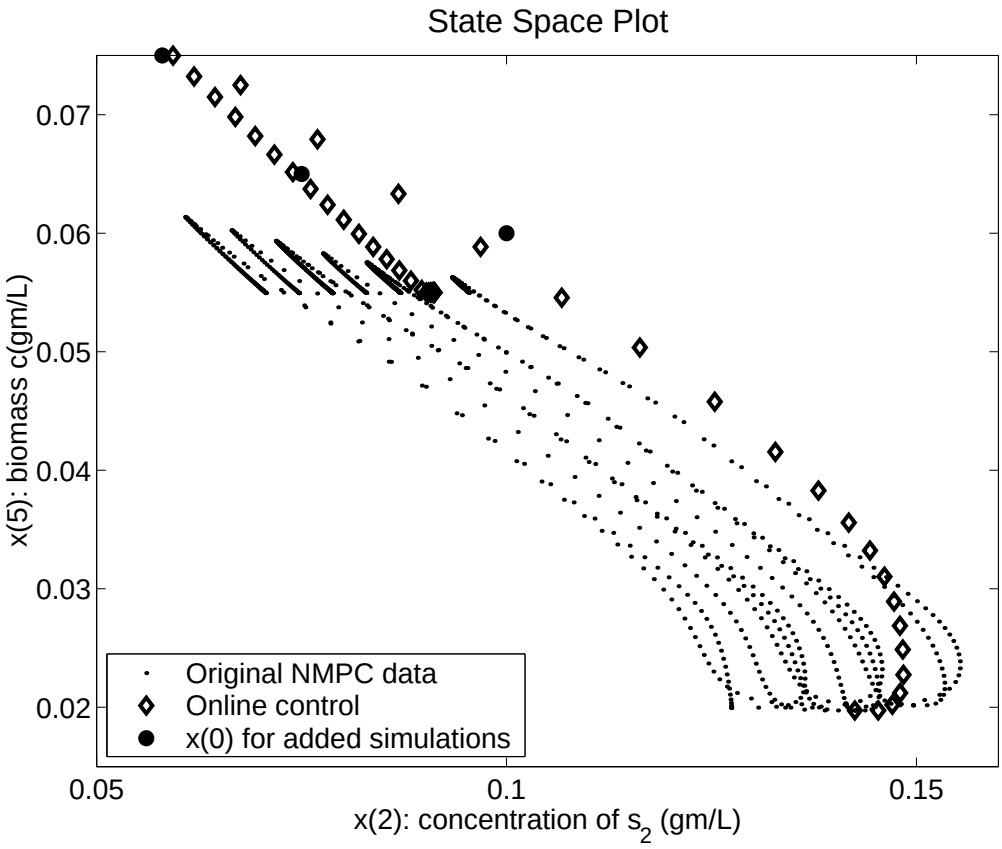

Figure 51: State space plot of states visited during online implementation ($\diamond$). The data from slMPC control law used for training the cost approximation is shown as dots. Extrapolation to the unvisited states during online implementation is likely to be the cause of overshoot.

law, both in performance of the controller as well as significant reduction in computation time. Corresponding state space plot is shown in Figure 53. Dots represent data points used for obtaining the cost-to-go function approximation. Diamonds represent the points visited during online implementation of the simDP control law. Comparing this with the previous state space plot (Figure 51), it is evident that by artificially constraining the control moves to the visited region, one obtains improved controller performance. Also note that the online optimization does not select the best among the suboptimal trajectories. Instead, it gives a trajectory which results from interpolating in the region defined by these suboptimal trajectories.

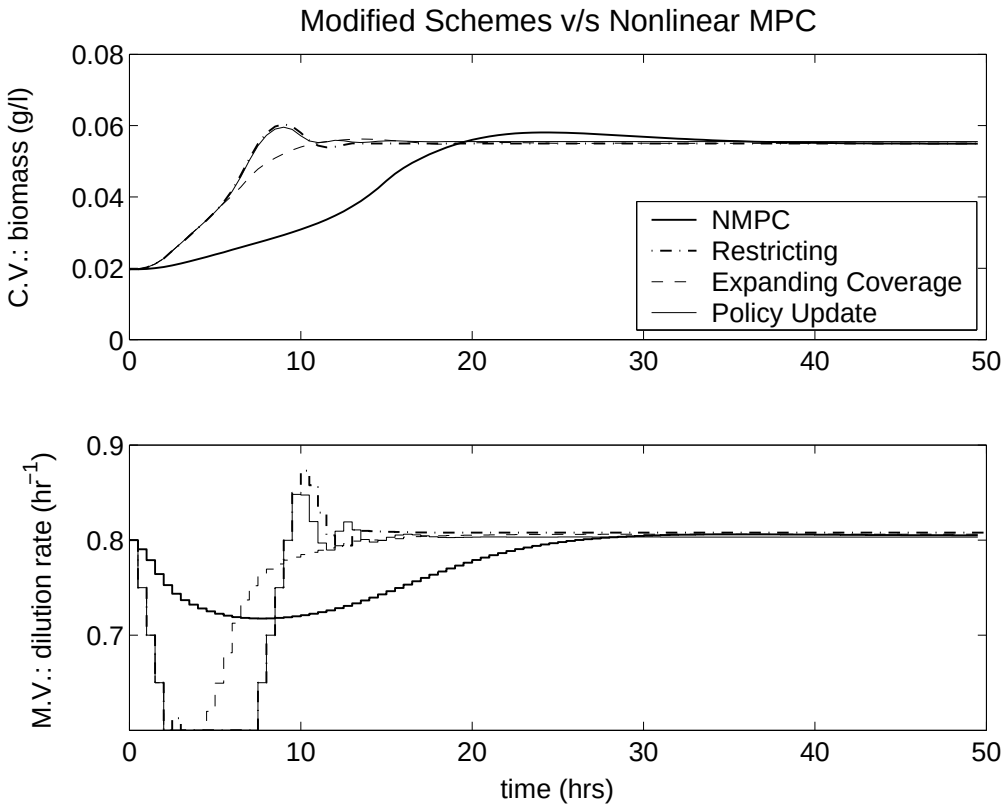

Figure 52: Performance of the various simDP schemes (thick line: slMPC)

6.3.4.2 Increasing data coverage through additional simulations

In the previous method, we sought to restrict our controller to the visited regions of the state space. An alternative solution is to increase the coverage of the training data through additional simulations of the initial slMPC control law. Additional data is obtained from some more simulations of the slMPC law and value iteration is performed again.

During online implementation of "unmodified" simDP algorithm, the controller drove the system to the unvisited region of the state space (dots in Figure 51 represent the visited region, rest is unvisited). Three such points — shown as solid discs in Figure 51 — were selected from the unvisited region. These represent $x(0)$ values for the additional simulations. We applied the original slMPC control, as before, for all the four values of s_{2f}. These 12 additional simulations resulted in 880 more data points. Value iteration was carried out again for all the 2088 (including 1200 from the previous case) data points.

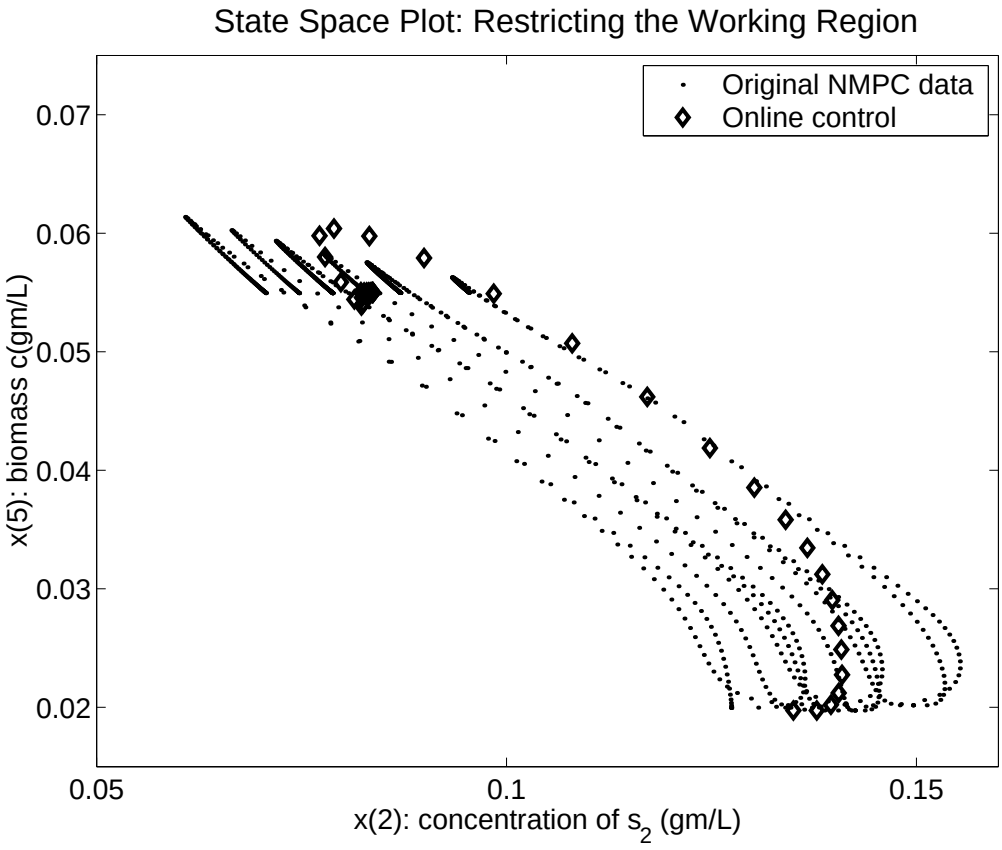

Figure 53: State space plot of states visited during online implementation when simDP was restricted to the visited region of the state space.

In this case, the Bellman equation converged in 4 iterations. The converged cost approximation was used in online control. Numerical and visual comparison indicate that this modification of simDP algorithm also results in highly improved control performance. In Figure 54, asterisks represent the 888 data point added to the original 1200 data (shown as dots). The resulting cost-to-go function approximation in this case is valid over a larger region. With these additional data points, the controller successfully avoids overshoot by identifying it as a suboptimal performance.

6.3.4.3 Generalized Policy Update

This technique is used to increase the coverage of the state space as in section 6.3.4.2. Unlike the previous scheme that requires addition of new data points and performing the entire set of iterations all over again, new points are added through policy update performed

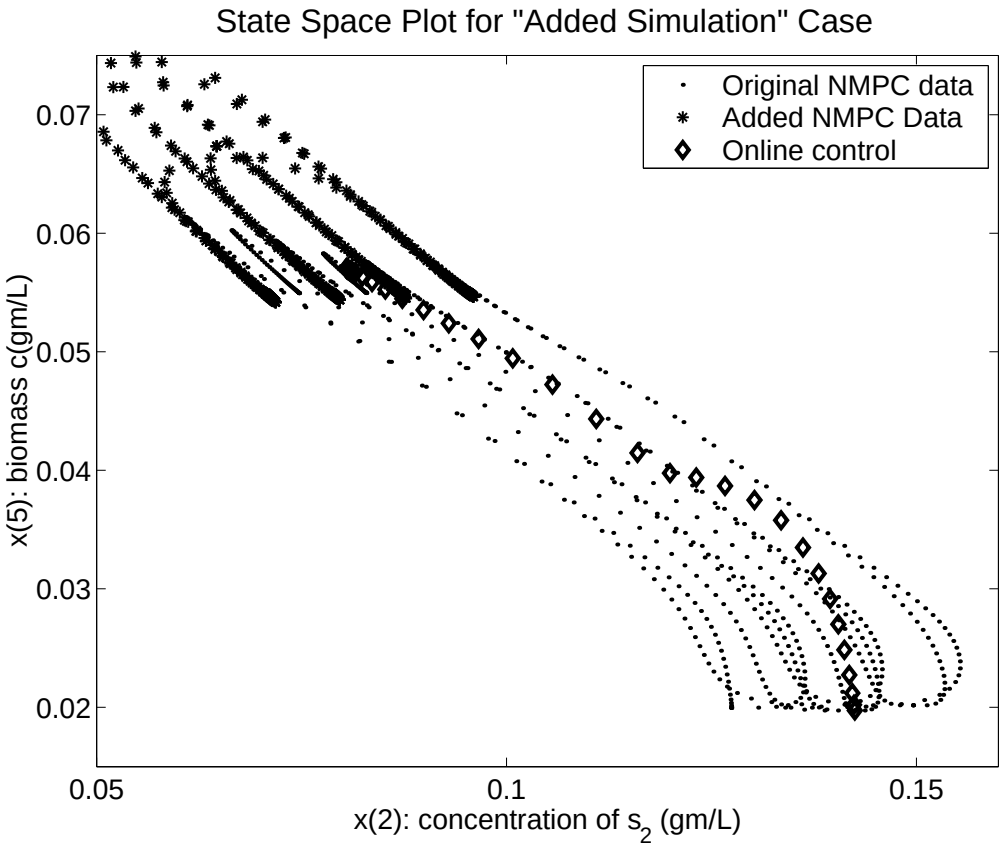

Figure 54: State space plot of states visited during online implementation ($\diamond$) with additional training data (*)

within the value iteration loop. The need for this generalized policy update arises because in real situations, we may not want to wait for deteriorated behavior of the controller to direct additional simulations of the initial slMPC law. Moreover, the above method involves redoing the entire value iteration procedure all over again.

The central theme of generalized policy update is to add more data points within the value iteration loop itself through closed loop simulation of the current suboptimal controller (J^i). In value iteration, the improved cost-to-go value $J^{i+1}(x)$ is obtained directly from the minimization step by using Bellman equation (59) as an update rule. Alternatively, policy iteration seeks to improve the policy by implementing it and evaluating the cost-to-go with this policy, instead of just updating the cost-to-go value [12]. In this sense, policy update can be considered a hybrid between value iteration approach and policy iteration. During

the value iteration procedure, solving the corresponding equation (62) for each of the 1200 states ($x(k)$) gives us an improved cost-to-go (J^{i+1}) and input argument $u(k)$. If this input move was to be implemented, the system would reach a particular state $x^i(k+1|k)$ (which means the state reached by implementing control move calculated using cost-to-go function approximation $\tilde{J}^i$ at time k). If this state lies in the unvisited region, following procedure is implemented

- Current state is initialized as $x_0^i = x^i(k+1|k)$
- Control move is computed as $u_0^i = \arg\min[\phi + \tilde{J}^i]$. Implementing this move results in $x_1^i = F_h(x_0^i, u_0^i)$
- If x_1^i lies in the visited region, cost-to-go at this state is given by the neural network $\tilde{J}^i(x_1^i)$. Otherwise, control moves u_1^i, u_2^i, $\ldots u_{Ni}^i$ and corresponding states x_2^i, x_3^i, $\ldots x_{Ni+1}^i$ are computed according to the steps described above until x_{Ni+1}^i lies in visited region or reaches the set point.
- Cost-to-go values J_j^{i+1} for state x_j^i are given by $J_j^{i+1} = \sum_{l=j+1}^{Ni} \{\phi(x(l), u(l))\} + \tilde{J}^i(x_{Ni+1}^i)$

The cost-to-go value is computed and this state is added to the original 1200 states. This is done for each state $x^i(k+1|k)$ that lies in the unvisted region. In other words, policy update and cost-to-go evaluation is carried out within the value iteration loop, hence the name. Then, a new functional approximator $\tilde{J}^{i+1}$ is fitted and (equation 62) is solved again for the expanded data set. This is done iteratively until convergence. Iteration is said to be converged if sum of absolute error was less than 5% and number of data points added during the iteration was less than 6 (0.5% of 1200).

In this case, the Bellman equation and policy update converged in 7 iterations. Numerical and visual comparison (Table 14, Fig. 52) indicate that this modification of simDP algorithm also results in highly improved control performance. Extrapolation to the unvisited states is avoided due to increased coverage by adding 195 data points (Fig. 55). We believe that this method is more rigorous and general than the previous two methods.

The optimal path may lie outside the region visited by the initial slMPC control law, which would not be considered if we were to restrict the online controller by gridding. Likewise, we do not need deteriorated behavior of the controller to direct additional simulations of initial slMPC controller to increase the data coverage, which may or may not work in a generalized scenario.

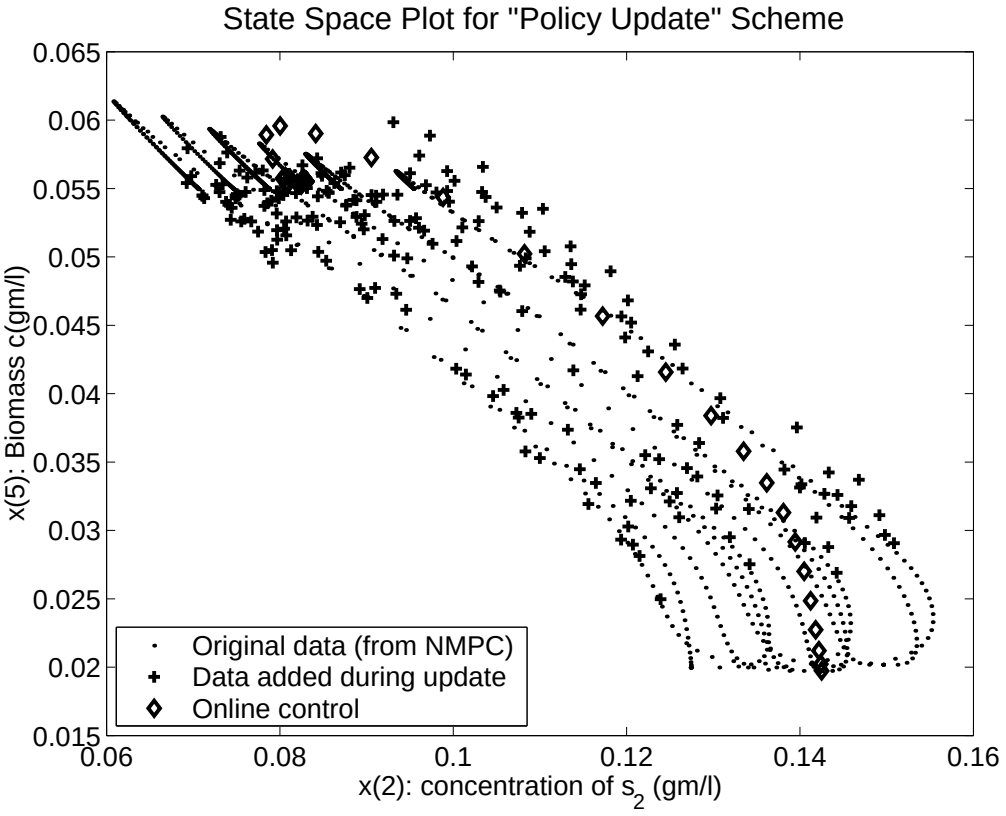

Figure 55: State space plot of states visited during online implementation of the "policy update" simDP scheme($\diamond$). Also shown are the data from the original suboptimal slMPC ($\cdot$) and the data points added during policy update ($+$). In all, 195 data points were added in the 7 iterations performed until convergence of cost-to-go function.

6.3.5 Comments

In this paper, we suggested a Simulation-Approximation-Evolution (simDP) strategy based on ideas from Neuro-Dynamic Programming and Reinforcement Learning literature, to control a continuous reactor with multiple steady states — a highly nonlinear control problem.

Traditional linear controllers are usually inadequate for such systems as the system dynamics may vary drastically at the different steady states and during transition from one type of steady state to another. The suggested simDP method transfers the computational burden of solving multistage nonlinear optimization problem to offline; instead, an equivalent single stage optimization is solved online — through the use of cost-to-go function approximation.

In Table 14, the four simDP algorithms suggested differ from each other in several aspects. The basic method as well as gridding method uses only the original 1200 point for computing cost-to-go approximation. In the other two methods, additional data points are added to increase the coverage of state space.

The different number of hidden nodes in the cost-to-go neural network suggests that the structure of cost approximation is different in the four simDP cases considered. The minimum number of hidden nodes that gave a mean square error (MSE) of 10^{-3} was selected (see Table 14).

6.4 Concluding Remarks

Application of general ideas from NDP and RL literature to improving performance of MPC for a highly nonlinear problem of steady state switching in a microbial reactor was considered. The results indicate that the simDP scheme developed provides a promising framework for nonlinear optimal control in a computationally amenable way. The proposed controller does not select the best amongst original suboptimal trajectories; instead it seems to interpolate in the region defined by these trajectories, directed by cost-to-go function approximation, to yield substantially improved performance. An important outcome of this work is that while the method shows great promise, one needs to be careful in using the cost-to-go function approximation. In this study, extrapolation to previously unvisited states resulted in poor performance of the "vanilla" simDP controller. Three different modifications were suggested. We are, however, biased towards the *generalized policy update* as a more systematic method of increasing the coverage and searching for more optimal policy even in the regions unvisited by the original suboptimal controller.

This also points to an important function initial suboptimal strategy plays. We think

that it is not necessary for initial suboptimal strategy to be close to optimal — Bellman iterations will take care of this issue. However, it is imperative that the suboptimal strategy covers all relevant portions of the state space. Spanning relevant parts of state space becomes increasingly difficult with an increase in state dimension. Hence Principal Component Analysis (PCA), Self-Organizing Maps (SOM) or other feature extraction schemes may be used to reduce the state dimension or better identify relevant regions of the state space.

CHAPTER VII

ON THE CHOICE OF COST APPROXIMATOR AND ITERATION ALGORITHM IN SIMDP

Key issues in implementation of simulation-based Dynamic Programming (sim-DP) are the addressed in this chapter. First, a comparison between a parametric "global" approximator like neural network and a local averager with respect to convergence during offline learning and online performance is presented. The local averager shows stable learning of the cost function and robust performance during online control. Next, the convergence properties of policy iteration are empirically investigated. We demonstrate that policy iteration requires lesser iterations than value iteration to converge, but requires more function evaluations to generate cost-to-go approximations in the policy evaluation step. Two different alternatives to policy evaluation, based on iteration over simulated states and simulation of improved policies are presented. We then demonstrate that the λ-policy iteration method, with $\lambda \in [0, 1]$, is a tradeoff between value and policy iteration. Finally, the issue of exploration to expand the coverage of the state space during offline iteration, and its impact on online performance is also considered.

7.1 Introduction

Dynamic Programming (DP), introduced by Bellman [8], generated a lot of interest as it provides a theoretically sound framework for solving multi-stage dynamic optimization problems. Such stagewise optimization is encountered in optimal control problems, Markov Decision Processes, planning and scheduling, etc. DP aims to characterize the optimal solution to the dynamic optimization in the form of "cost-to-go" or "value" function, which expresses the desirability of any state x in the state space with respect to the long-term performance that can be achieved.

Simulation-based approximate Dynamic Programming (sim-DP) methods were introduced in the previous chapter to alleviate the curse of dimensionality in conventional DP. According to Sutton and Barto [104], the main idea of these methods is that they aim to obtain good *suboptimal* solutions to the DP problem, starting with suboptimal policies and suboptimal cost-to-go function, iteratively using each to improve the other. These methods use selective sampling through simulations to identify relevant subset of the state space, a function approximator to interpolate the cost-to-go values in this "visited" subset, and an iterative learning algorithm to improve the cost-to-go values. This chapter highlights the importance of these three components: function approximator, iteration algorithm and the state space coverage.

First, the issue of choosing an appropriate algorithm is addressed. Reinforcement Learning literature has seen a chequered history with the use of an appropriate approximator: there have been successful implementation of neural networks as function approximators in difficult problems such as TD-gammon [107], as well as notable failures in relatively simpler cases [17]. Boyan and Moore [16] showed that the iteration algorithms that use parametric approximations of the value functions fail to converge stably in several cases. Gordon [41] presented a class of function approximators with "non-expansion" property to guarantee stable learning of the cost-to-go function, whereas Ormoneit and Sen [84] adopted non-parametric Kernel-based averaging schemes that always converges to a unique solution. We extend this result to systems with continuous state and action spaces, and show that local averagers provide stable offline learning and improved online performance. An interested reader is referred to [66] and [67] for more details on this issue.

The next issue we address is the offline convergence properties of iteration algorithms. Of particular interest in this chapter is simulation-based approximate policy iteration. Policy iteration, often attributed to Howard [51], generates an improving sequence of policies from the corresponding cost-to-go functions, followed by updating the cost-to-go function by evaluating the generated policy. Our aim is to compare the policy iteration approach with value iteration. Recently, Santos and Rust [93] proved that under certain regularity conditions, policy iteration shows superlinear or quadratic rate of convergence. In contrast,

value iteration has only a linear rate of convergence. De Farias and Van Roy [24] argued that approximate value iteration need not converge under some cases. Based on their studies of temporal difference learning (which, like policy iteration, iterates in the policy space), they suggested a variation of approximate value iteration that is guaranteed to converge.

Clearly, approximate policy iteration algorithm converges faster than value iteration. However, the policy evaluation step itself is an iterative step, which sometimes fails to converge in a reasonable amount of time [12]. Unlike policy improvement, each iteration within policy evaluation does not require any minimization and is therefore computationally less intensive that policy improvement. We investigate this tradeoff between lesser number of iterations (and therefore lesser number of minimizations) and a greater number of iterations during policy evaluation. The results of convergence behavior observed in Markov Decision Processes are extended to control of systems with continuous state and action spaces. We also investigate the following alternative ways to perform policy evaluation:

- Computing cost-to-go iteratively over the data points determined by the initial sub-optimal control simulations (iter-PI)
- Through simulation of the improved policies generated during the policy iteration. Cost-to-go values are directly calculated as the sum of individual costs until steady state (sim-PI)
- Using temporal difference based λ-policy iteration (λ-PI of Bertsekas and Ioffe [11])

Finally, we allude to the issue of state space coverage. As the cost-to-go approximators are built with data occupying a limited subset of the state space, we need to avoid extreme extrapolations during online control. Additionally, policy iteration can be extended to provide exploration of the unvisited regions of the state space during offline learning phase, through input dithering.

7.2 Background and Key Issues

The background for Dynamic Programming (DP) is briefly summarized and key issues in implementation of sim-DP are presented in this section. The main aim of DP is to obtain

an optimal value or cost-to-go function for any state $x \in \mathcal{X}$. The optimal cost-to-go is shown to satisfy the Bellman equation (59). The Bellman equation can be represented in the form of DP operator T as

$$J^* = TJ^* \tag{68}$$

Blackwell [13] was amongst the first authors to rigorously prove that the DP operator T is a contraction mapping. This is an important result as it guarantees convergence to a unique solution J^* through the use of iterative algorithms.

The conventional DP suffers from the curse of dimensionality as the cost-to-go values need to be computed for each state in the state space. Sim-DP alleviates this curse of dimensionality by restricting the iterative computation of cost-to-go to a limited subset of state space, and using a function approximator to interpolate within this subset. The three key issues, viz. the approximator, iteration algorithm and state space coverage are discussed here.

7.2.1 Cost-to-go function approximator

The choice of cost approximator is an important issue in simDP methods. The cost approximator should provide good interpolation within the visited subset of the state space, and avoid over-extrapolation of cost-to-go. At the same time, it should also allow for guarded exploration beyond the visited subset. Two main classes of approximators compared are global parametric approximator, and a local averager.

7.2.1.1 Global approximator

A global approximator uses parametric representation to approximate the cost-to-go as a function of state variables, such as linear combination of basis vectors, neural networks, etc. We chose a feedforward neural network with one hidden layer as the first choice. The neural network takes in augmented state as the inputs and the cost-to-go as the output. The network is trained after each value iteration and after each policy evaluation using Levenberg-Marquardt method (using the `trainlm` command in the Matlab neural network toolbox).

7.2.1.2 Local averagers

Local averagers are memory-based methods that find cost-to-go values for a query point as a weighted average of its neighboring points. Let x_q be a query point and $\hat{x}_1, \ldots, \hat{x}_i, \ldots$ be the neighboring points in the memory. In this chapter, we consider two different local averagers.

The first averager is the k-nearest neighbor. Here, the cost-to-go value of a query point x_q is approximated as the distance weighted mean of its nearest neighbors $\hat{x}_1, \ldots, \hat{x}_k$. For all the examples, we chose $k = 4$. The cost-to-go value of x_q is given by

$$\tilde{J}(x_q) = \frac{\sum_{i=1}^{k} \gamma_i J(\hat{x}_i)}{\sum_{i=1}^{k} \gamma_i} \qquad \text{where } \frac{1}{\gamma_i} = [x_q - \hat{x}_i]^T W [x_q - \hat{x}_i] \tag{69}$$

Here W is a feature weighting matrix. If $W = I$, the γ_is become inverse of the square of the euclidian distance between the two points. Typically, we choose the feature weight matrix as $W = \texttt{diag}[w_1, \ldots, w_k]$. This allows us to emphasize the state variables that are more important than others.

A Gaussian kernel-based local averager is the other approximator used in this work. The cost-to-go value for x_q is obtained as a weighted average of all the points lying within a pre-defined threshold distance from it. Let $\hat{x}_i$, $i = 1, \ldots, N$ be the N data points within a hyper-sphere of radius ρ around the query point x_q. The cost-to-go for x_q was then defined as as:

$$\tilde{J}(x_q) \quad = \quad \begin{cases} \dfrac{\sum_{i=1}^{N} K_\lambda(x_q, \hat{x}_i) J(\hat{x}_i)}{\sum_{i=1}^{N} K_\lambda(x_q, \hat{x}_i)} & \text{if } N > N_{min} \\ \\ J_{max} & \text{otherwise} \end{cases} \tag{70}$$

where the kernel is given by

$$K_\lambda(x_q, \hat{x}_i) = \exp\left(-\frac{\|x_q - \hat{x}_i\|_2^2}{\lambda^2}\right) \tag{71}$$

The design parameters ρ, N_{min}, λ and J_{max} are chosen considering the coverage of state space. If there are less than N_{min} visited data points within the threshold distance ρ, the

query point is assigned a large cost. Thus, the Gaussian kernel-based averager prevents extrapolation.

7.2.2 Iteration algorithm

Value iteration uses the Bellman equation (59) as an update rule to sequentially improve the cost-to-go values, starting from a suboptimal cost-to-go $J^0(x)$. Therefore, value iteration is also called "the method of successive approximations." In contrast, policy iteration is a two step procedure. First, policy improvement is used to find an improving policy $\mu^{i+1}(x)$ using cost-to-go function $J^i(x)$ according to:

$$\mu^{i+1}(x) = \arg\min_u \left\{ \phi(x,u) + J^i\left(f_h(x,u)\right) \right\} \tag{72}$$

The second *policy evaluation* step involves following the policy $\mu^{i+1}(x)$ to compute cost-to-go function. The updated cost-to-go function is given by the following implicit equation

$$J^{i+1}(x) = \phi(x,\mu^{i+1}(x)) + J^{i+1}\left(f_h(x,\mu^{i+1}(x))\right) \tag{73}$$

The details of the two algorithms are presented in Appendix B.

Puterman and Brumelle [88] showed the equivalence between policy iteration and Newton's method. They proved that policy iteration has super linear or quadratic rate of convergence, while value iteration has only a linear rate of convergence. Thus, policy iteration converges faster than value iteration. Additionally, due to the super linear rate, policy iteration converges to the exact solution, whereas value iteration converges asymptotically. Santos and Rust [93] extended this result to approximate DP. While policy iteration converges faster, the policy evaluation itself is an iterative step. This represents a tradeoff between lesser number of iterations (which means lesser minimization steps) and a larger number of function evaluations for policy evaluation step. The following three alternative choices for policy evaluation are considered

- **Iterative evaluation:** In the policy iteration algorithm mentioned above (and in Appendix B), we perform the iterations over all data points in the memory using one-step ahead simulation of the improved policy. The policy evaluation is then

iteratively performed using Eq. (64). We call this method iterative evaluation-based policy iteration or iter-PI.

- **Simulation-based evaluation:** An alternative method is to start with some selected initial points and perform simulations using the updated policy $\mu^{i+1}(x)$ until steady state. Instead of generating a single successor state ($f_h(x, \mu(x))$), a series of states from the selected initial points to the setpoint are generated. The policy evaluation for the simulated policy is simply the sum of single stage costs until steady state:

$$J^{i+1}(x_k^{sim,i+1}) = \sum_{l=1}^{\infty} \phi(x_{k+l}^{sim,i+1}, u_{k+l}) \tag{74}$$

  Here, the superscript ^{sim} is used to identify that the states visited during the simulations. We call this method sim-PI.

- **λ-Policy Iteration** The λ-policy iteration was proposed by Bertsekas and Ioffe[11] as a method to accelerate the policy evaluation step by introducing a "discount factor" $\lambda \in [0 \quad 1]$ that does not alter the cost-to-go. In λ-policy iteration, we use the temporal difference for the state transition $x_k \rightarrow x_{k+1}$ to modify policy evaluation as

$$J^{i+1}(x_k) = J^i(x_k) + \sum_{m=0}^{\infty} \lambda^m d(x_{k+m}) \tag{75}$$

  This method, which was described in the previous chapter, reduces to pure value iteration for $\lambda = 0$ and policy iteration for $\lambda = 1$. Thus, one can view this as some form of "interpolation" between value and policy iteration.

7.2.3 Coverage and exploration

One of the important features of simDP is that the iterations are performed in a restricted region of the state space defined by the simulation data $\mathcal{X}^{sim}$. Often, the coverage of the state space may not be adequate, and more simulations may be required to increase the coverage. There are two alternative ways to do this:

- After convergence, the cost-to-go function can be used to simulate online performance. Additional states visited can guide additional suboptimal simulations to generate more

data. The learning iterations can be performed again with the additional data. This case was considered in section 6.3.4.2.

- Cautious exploration of the state space can be performed with policy iteration, either by allowing the controller to visit previously unvisited regions or by adding extraneous dithering signal.

7.3 Linear Quadratic Control Example

Before moving to sim-DP examples for offline convergence and online performance of policy iteration, a comparison of value and policy iteration for a linear unconstrained system is presented in this section. Consider the problem of regulating a linear unconstrained system

$$x_{k+1} = Ax_k + Bu_k \tag{76}$$

to the origin. The single stage cost for this system is $\phi(x,u) = x^TQx + u^TRu$. An analytical solution of DP exists in the form of Linear Quadratic Regulator (LQR). It can be proved that the optimal cost-to-go function is a quadratic function of the system state [10]. Hence, the cost-to-go function at i^{th} iteration is denoted as

$$J^i(x) = x^TS^ix \tag{77}$$

where i is the iteration index. The infinite-horizon problem of (53) can be equivalently represented as

$$\min_{u_k}\left\{\left[x_k^TQx_k + u_k^TRu_k\right] + x_{k+1}^TS^\infty x_{k+1}\right\} \tag{78}$$

7.3.1 Analytical Solutions

We can show that the input that minimizes the above objective (78) is given by

$$\begin{aligned} u_k &\triangleq -L\cdot x_k \\ &= -(B^TS^\infty B + R)^{-1}B^TS^\infty A\cdot x_k \end{aligned} \tag{79}$$

Value iteration involves updating the cost function $J = x^TSx$ with the minimizer function obtained by solving Eq. (78) above. Thus,

$$J^\infty(x_k) = \min_{u_k}\left[x_k^TQx_k + u_k^TRu_k\right] + J^\infty(x_{k+1})$$

$$x_k^T S^\infty x_k \quad = \quad \min_{u_k} \left[x_k^T Q x_k + u_k^T R u_k \right] + x_{k+1}^T S^\infty x_{k+1}$$

Substituting u_k from Eq. (79), the solution of above equation is given by Ricatti Difference Equation (RDE) as in the famous LQR problem:

$$S^\infty = A^T S^\infty A + Q - A^T S^\infty B [B^T S^\infty B + R]^{-1} B^T S^\infty A \tag{80}$$

Therefore, each value iteration is equivalent to solving one step of this Ricatti Difference Equation.

Policy iteration is a two-step procedure. First, the improved policy is computed according to Eq. (79) as $\mu^{i+1}(x_k) = -L^{i+1} \cdot x_k$ given current estimate of cost-to-go function $J^i(x) = x^T S^i x$. Next, policy evaluation (Eq. 73) is used to compute the updated cost-to-go function as:

$$x_k^T S^{i+1} x_k = x_k^T [Q + L^{(i+1)T} R L^{i+1}] x_k + x_{k+1}^T S^{i+1} x_{k+1} \tag{81}$$

Since $\mu(x_k) = -L^{i+1} \cdot x$, we can write the successor state value as

$$x_{k+1} = \left[A - B L^{i+1} \right] x_k$$

Thus, policy evaluation is equivalent to solving a discrete Lyapunov equation:

$$Y S^{i+1} Y^T - S^{i+1} + Z = 0 \tag{82}$$

$$\begin{aligned} \text{where} \qquad & Y = [A - B L^{i+1}]^T \\ \text{and} \qquad & Z = [Q + L^{(i+1)T} R L^{i+1}]. \end{aligned} \tag{83}$$

sim-PI was introduced as an alternative method, which generates new cost-to-go function through simulations of the policy $\mu^{i+1}(x)$. The policy improvement step remains the same, whereas we apply policy evaluation for the ∞-horizon cost as:

$$J^{i+1}(x_k) \quad = \quad \sum_{m=0}^{\infty} \phi \left(x_{k+m}, \mu^{i+1}(x_{k+m}) \right) \tag{84}$$

$$\quad = \quad \sum_{m=0}^{\infty} x_{k+m}^T \left[Q + L^{(i+1)T} R L^{i+1} \right] x_{k+m}, \tag{85}$$

Since $\mu(x_k) = -L^{i+1} \cdot x_k$,

$$x_{k+1} = [A - BL^{i+1}]x_k$$

$$x_{k+2} = [A - BL^{i+1}]x_{k+1} = [A - BL^{i+1}]^2 x_k$$

$$\vdots$$

$$x_{k+m} = [A - BL^{i+1}]^m x_k$$

Using this, we get

$$x_k^T S^{i+1} x_k = x_k^T \left\{ \sum_{m=0}^{\infty} Y^{Tm} Z Y^m \right\} x_k \tag{86}$$

where Y and Z are as defined previously in Eq. 83. This reduces to the same Lyapunov equation (87) as before.

In **λ-Policy Iteration**, policy improvement step is performed as before (Eq. 79). However, policy evaluation is modified using a discount factor λ, which does not alter the cost-to-go structure for a given policy. Policy evaluation reduces to solving a different Lyapunov equation

$$\tilde{Y} S^{i+1} \tilde{Y}^T - S^{i+1} + Z = 0 \tag{87}$$

$$\begin{aligned} \text{where} \quad & \tilde{Y} = \lambda^{0.5}[A - BL^{i+1}]^T, \\ & \tilde{Z} = \lambda[Q + L^{(i+1)T} R L^{i+1} + (1-\lambda) S^{i,rd} \\ \text{and} \quad & S^{i,rd} = A^T S^i A + Q - A^T S^i B[B^T S^i B + R]^{-1} B^T S^i A \end{aligned}$$

$S^{i,rd}$ is the solution of one step of RDE, just like that given by Eq. (80). Proof is provided in Appendix C.

λ-policy iteration is a balance between value iteration and policy iteration:

- When $\lambda = 0$, $\tilde{Y} = 0$. Thus, Eq. (87) reduces to $-S^{i+1} + S^{i,rd} = 0$; ie. *0-policy iteration* is nothing but value iteration.
- When $\lambda = 1$, $\tilde{Y} = Y$ and $\tilde{Z} = Z$; ie. *1-policy iteration* is the standard policy iteration.

Comparing Eq. (87) with Eq. (82), we can see that λ-policy iteration involves the following modified policy evaluation step:

$$\begin{aligned} J^{i+1}(x_k) &= (1-\lambda)\left[\phi(x_k,u_k)+J^i(x_{k+1})\right] \\ &+ \lambda\left[\phi(x_k,u_k)+J^{i+1}(x_{k+1})\right] \end{aligned} \tag{88}$$

7.3.2 Numerical comparison

The following numerical example is used to illustrate the difference between convergence property of value and policy iteration:

$$A = \begin{bmatrix} 1 & 0 \\ 0.1 & 1 \end{bmatrix} \qquad B = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

The weighting matrices for one-stage cost were chosen as $Q = I_2$ and $R = 0.01$. The optimal S^* matrix was obtained using `dlqr` function in MATLAB. The matrix S^0 used to initialize value and policy iteration and the optimal S^* matrix are shown below:

$$S^0 = \begin{bmatrix} 1 & 0.1 \\ 0.1 & 1 \end{bmatrix} \qquad S^* = \begin{bmatrix} 1.117 & 1.062 \\ 1.062 & 11.522 \end{bmatrix}$$

Convergence criterion used was that the Frobenius norm of the difference between consecutive cost-to-go functions, $\|S^{i+1} - S^i\| \leq 0.001$. Value and policy iteration converged in 43 and 7 iterations respectively. The converged S^{43} and S^7 matrices shown in Table 15 indicate that the convergence of value iteration is asymptotic, while that of policy iteration is exact. However, policy evaluation itself is an iterative step. We did not use `dlyap` function in Matlab but iterated using

$$S^{i+1,j+1} = Y S^{i+1,j} Y^T + Z \tag{89}$$

to perform policy evaluation; it took 346 evaluations for just the first policy iteration to converge. This indicates a potential problem with policy iteration—policy evaluation can be very slow in converging.

Table 15: S^i values for unconstrained linear system

Value Iteration		Policy Iteration	
S^1	S^{43}	S^1	S^7
$\begin{bmatrix} 1.02 & 0.1 \\ 0.1 & 1.99 \end{bmatrix}$	$\begin{bmatrix} 1.117 & 1.061 \\ 1.061 & 11.518 \end{bmatrix}$	$\begin{bmatrix} 1.528 & 5.126 \\ 5.126 & 55.76 \end{bmatrix}$	$\begin{bmatrix} 1.117 & 1.062 \\ 1.062 & 11.522 \end{bmatrix}$

Table 16: λ-policy iteration schemes applied to the unconstrained linear system, for various values of λ.

λ	$\lambda = 0$	$\lambda = 0.05$	$\lambda = 0.25$	$\lambda = 0.5$	$\lambda = 0.75$	$\lambda = 0.95$	$\lambda = 1$
Iterations	43	41	34	25	16	8	7
PEval1*	–	4	6	11	24	99	346

*PEval1: number of policy evaluations required during the first policy iteration

Table 16 demonstrates that the λ-policy iteration is a balance between value iteration and policy iteration. As the λ value decreased to 0, more iterations are needed for cost-to-go function to converge, while the number of policy evaluations required for single policy iteration is reduced.

7.4 Numerical Examples

7.4.1 Constrained linear system

The first system considered for the comparison of value and policy iteration is a two-state constrained linear system, described by $y = 2\left(2s^2 + 3s + 2\right)^{-1}$. The system is sampled every $t_s = 0.1$ seconds. The resulting state space representation is

$$x(k+1) = \begin{bmatrix} 0.7326 & -0.0861 \\ 0.1722 & 0.9909 \end{bmatrix} x(k) + \begin{bmatrix} 0.0609 \\ 0.0064 \end{bmatrix} u(k) \tag{90}$$

The input is constrained within the values $-2 \leq u(k) \leq 2$. Bemporad et al. [9] developed the constrained LQR for this system using their parametric programming approach.

First, we applied a PI controller[1] ($K_c = 2$, $\tau_i = 1.2$, $|u(k)| \leq 2$) to this system. Regulation of the state to the origin was the control objective. Starting with 6 different x_0 values, we simulated the system until it reached the desired set point. A total of 420 data points were obtained using the PI controller, represented as policy $\tilde{\mu}^0$.

A neural network with two input nodes, four hidden nodes and one output node was used as the cost approximator. The iterations were said to have converged when the mean square error between the subsequent cost-to-go approximator was less than 0.01. The learning with neural network is non-monotonous. Hence, a mean square error criterion was used instead of using maximum relative error criteria. The results from using gaussian kernel-based averager were similar and are hence skipped for brevity.

It took nine iterations for value iteration to converge. Each iteration involved solving the minimization to get the updated cost-to-go value, followed by training the neural network. Policy iteration, on the other hand, converged in just three iterations. However, the total number of policy evaluations required were sixteen (8 in first, 5 in second and 3 in third policy iteration). Each policy iteration involves solving the minimization problem (policy improvement), which was solved only 3 times. However, after each policy evaluation, the neural network was trained again with the new cost-to-go values. Thus the computational requirement is higher for policy iteration.

The converged cost-to-go approximator was then used for online control. Figure 56 shows the online performance using cost-to-go function obtained by value and policy iteration methods. Both these methods show online performance similar to Bemporad's constrained LQR [9] and significantly better than the original suboptimal PI control. Another interesting result is that value iteration "over estimates" the cost-to-go values in the region where control actions hit the constraints. Figure 57 shows comparison between the prediction of cost-to-go approximators obtained by the two methods. This is attributed to the unstable learning of cost-to-go function with neural network. Such an "over estimation" was not observed with the local averager. The figure also shows that the cost-to-go surface is

[1]PI stands for Proportional Integral control. To avoid ambiguity, we use the terms iter-PI, sim-PI or λ-PI for policy iteration.

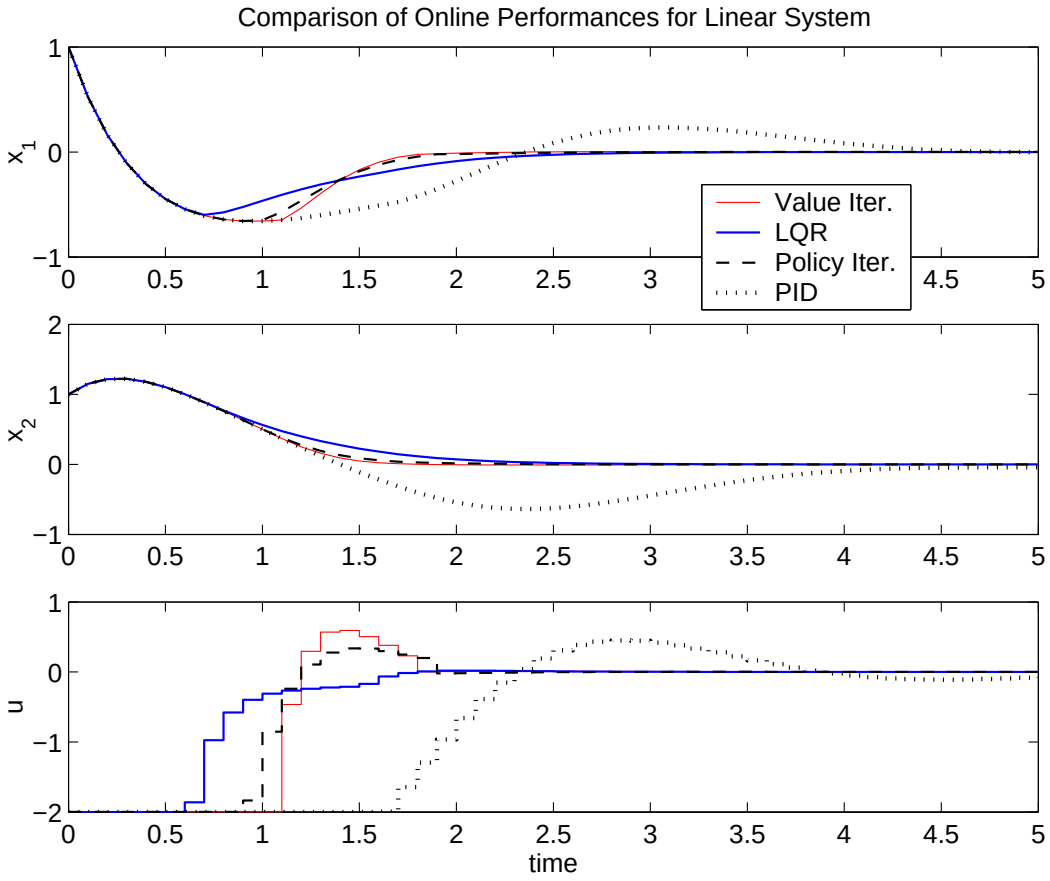

Figure 56: Online performance of value iteration (—) and policy iteration (– –) are significantly better than original PID control (· · ·). The online performance is comparable to the optimal LQR (—).

smooth. For such systems, although neural network shows unstable learning, the online performance is similar to that with a local averager.

7.4.2 Linear system with "soft" state constraints

An example of a system with "soft" state constraints is used for comparing local averager and a parametric approximator. In this example, we consider the problem of disturbance rejection for a linear system

$$y(s) = \frac{9.62}{s^2 + 2.4s + 5.05}u(s) + \frac{-19s - 57.3}{s^2 + 2.4s + 5.05}d(s) \tag{91}$$

with state and input constraints. The model originally refers to the control of engine rpm (y) using a bypass valve (u) in the presence of step disturbance in torque load (d). The

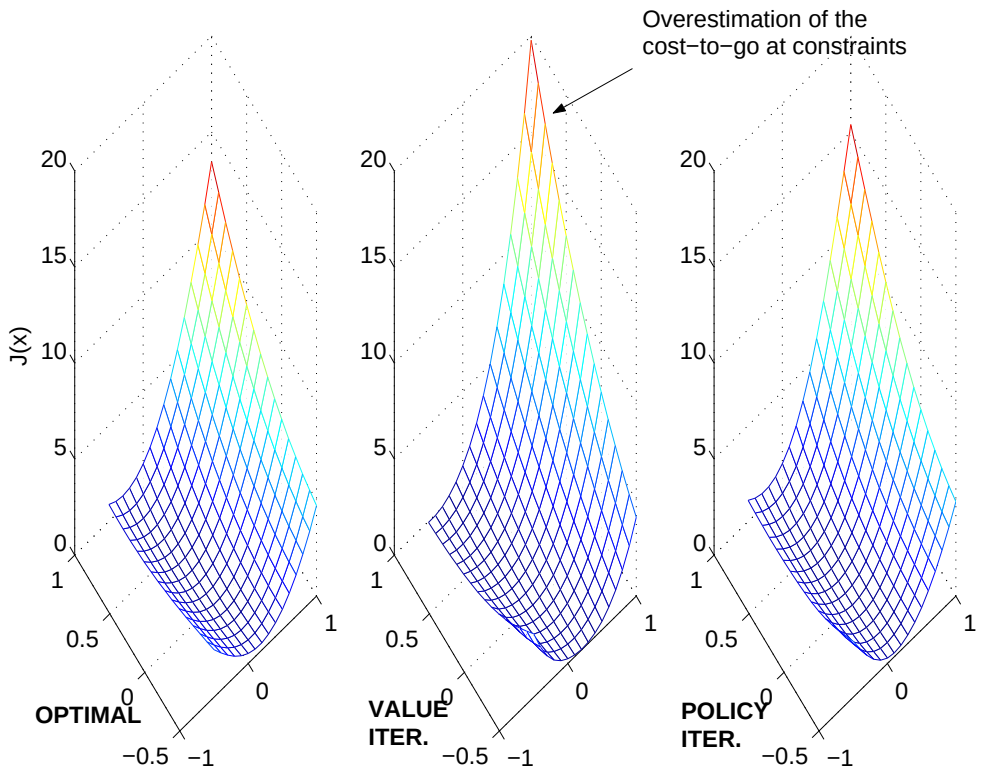

Figure 57: Optimal cost-to-go values and cost-to-go predictions by value and policy iteration. The overall structure of cost-to-go is obtained accurately. Value iteration overestimates the cost-to-go for states at the constraint.

system was sampled every 0.2 seconds, resulting in the following state space model:

$$
\begin{aligned}
x(k+1) &= \begin{bmatrix} -0.04223 & -0.3932 & -0.08433 \\ 0.04299 & 1.035 & -0.3984 \\ 0.08795 & 0.4352 & 0.4681 \end{bmatrix} x(k) + \begin{bmatrix} 0.06815 \\ 0.2912 \\ 0.6062 \end{bmatrix} u(k) + \\
&\quad \begin{bmatrix} -0.2026 \\ 1.68 \\ -0.468 \end{bmatrix} d(k) \qquad (92) \\
y(k) &= \begin{bmatrix} 0.2999 & -2.021 & 0.9494 \end{bmatrix} x(k)
\end{aligned}
$$

The pertinent constraints imposed were $-5 \leq u \leq 5$ and $-5 \leq y \leq 5$. The disturbances are assumed to be known or measured. For $d > 0.84$, the system cannot be controlled at the setpoint $y = 0$ with the given constraints. We looked at an interesting case of $d = 0.8$. Starting at the origin, no sequence of control actions $u(k)$ can be found to satisfy state constraints. Thus, constraint softening, by assigning larger cost penalty for constraint violation, was necessary.

Two different PI controllers ($K_c = 0.25$, $K_i = 0.15$; $K_c = 0.5$, $K_i = 0.08$) were used as initial suboptimal policy, with the input moves truncated to satisfy the constraint $|u| \leq 5$. The simulations were performed for two different starting points and four different d values. 75 data points were obtained from each of the 16 scenarios (2 controllers, 2 initial conditions, 4 disturbances). The weighting matrices for one-stage cost were chosen to be $Q = 1$ and $R = 0.04$. For the points where constraints are violated, one-stage cost were weighed 100 times (*i.e.* $Q = 100$, $R = 4$ if $|y(k)| > 5$). For this constraint softening case, the cost-to-go function has a very stiff structure. This is because some regions of the state space that violate (or are likely to violate) state constraints are associated with very high cost-to-go, while other regions of the state space are associated with lower cost-to-go.

7.4.2.1 Choice of the cost approximator

The cost-to-go was expressed as a function of the augmented state, consisting of the system state, disturbance and deviation from the setpoint; i.e. x = $[x_1,\ x_2,\ x_3,\ d,\ (r-y)]^T$. The system response depends on the state as well as disturbance. Deviation from set point was added to the augmented state for integral action of the controller. Two different approximators were used to obtain cost-to-go as a function of the augmented state:

- A multi-layer perceptron neural network (MLP) with 7 hidden nodes, and
- k-nearest neighbor (kNN), with $k = 4$ neighbors. As the disturbance (d) and error ($r - y$) are more critical than the three system states, a feature weighting matrix $W = \texttt{diag}[1,\ 1,\ 1,\ 6,\ 10]$ was used to compute the approximation according to Eq. 69.

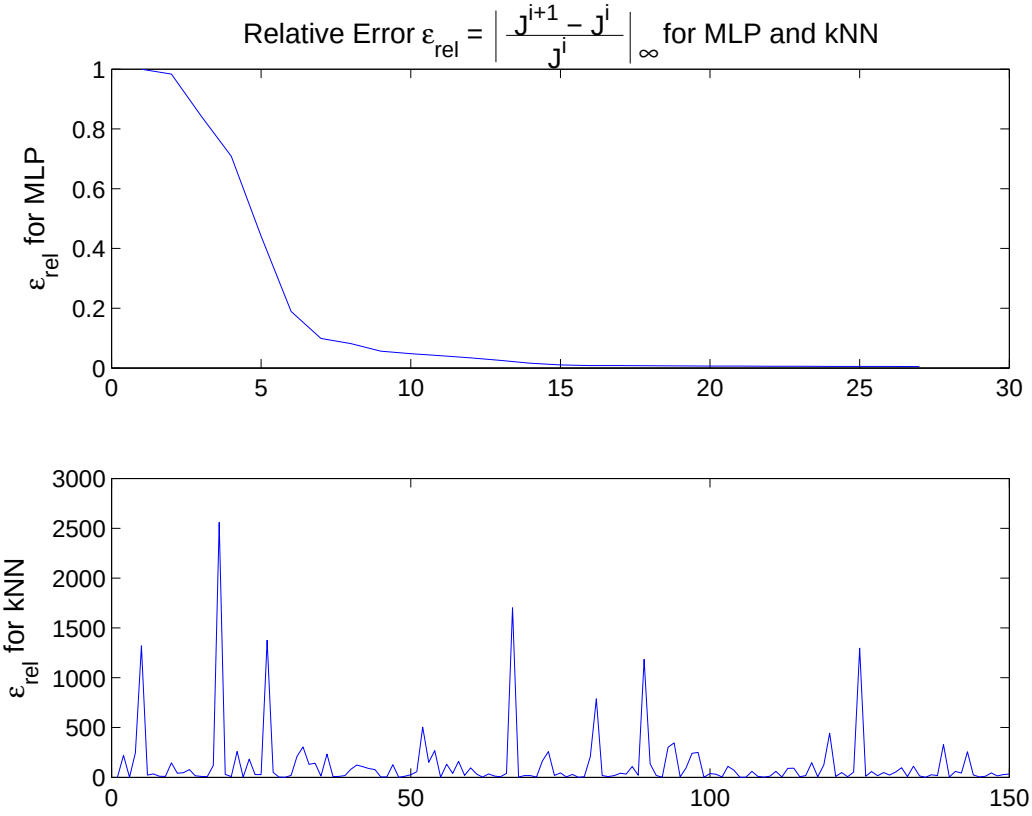

Figure 58: Relative error for learning using MLP and kNN. The learning in case of MLP is unstable while it is stable and monotonic for kNN.

The neural network was unable to provide a good approximation of the stiff cost-to-go function. The cost-to-go function did not converge even after 150 iterations. As shown in Figure 58-a, the learning with MLP is unstable. On the other hand, the learning of cost-to-go function with kNN was stable and the relative error monotonically decreased with increasing number of value iterations (Figure 58-b). The cost-to-go function was said to be converged when the infinity norm of the relative error between subsequent value iterations was less than 0.005. Value iteration converged in 27 iterations with kNN.

The accuracy of the cost-to-go function was tested by using them for online control. None of the 150 trained neural networks were able to control the system. The dash-dot line in Fig. 59 shows the performance of the best MLP cost approximator. The online performance of simDP with kNN (solid line in Fig. 59) is comparable to that of the truly optimal $\infty-$horizon MPC. The specific plots are for $d = 0.8$, wherein constraint softening

is required as at least one point violates the constraint. Note that $d = 0.8$ is a "new point", i.e. this condition was not used in learning the cost function.

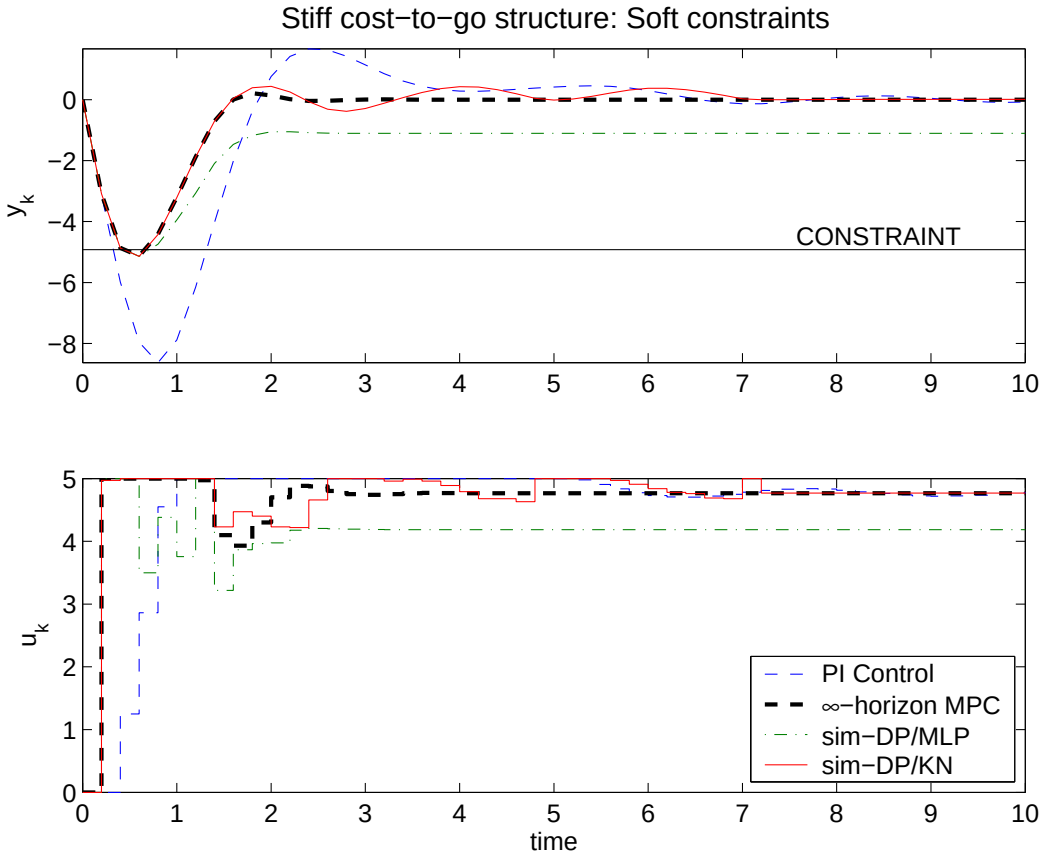

Figure 59: Online performance of sim-DP with the two approximators is compared with the optimal ∞-horizon control and the original suboptimal PI control

7.4.2.2 Convergence of policy iteration

We performed VI and iter-PI, this time using the Gaussian kernel-based averager, for this example. The algorithms were said to be converged when the absolute error between two consecutive iterations satisfied

$$e_{abs} \triangleq \max_{k;k=1,\ldots,1200} |J_k^{i+1} - J_k^i| < 0.001$$

Table 17 shows the comparison between VI, iter-PI and sim-PI. VI required 17 iterations and a computation time of 6641 seconds to converge during offline learning. On the other hand, iter-PI took 5 iterations and a total of 53 policy evaluations. The policy evaluation

Table 17: Convergence properties during offline learning of cost-to-go function for linear system with soft constraints

	Value Iteration	iter-PI	sim-PI
Iterations	17	5	6
Policy Evaluations	—	53	—
Offline computation time (s)	6641	2012	986

step requires computation of cost-to-go values for all visited states under the new policy. However, it does not require solving of a minimization problem. Hence, the overall computational requirement for policy iteration was lower than value iteration. For this example, sim-PI converged in 6 iterations. The significantly lower computation time was because simulation of this linear system, starting from an arbitrary x_0 to the setpoint is very fast.

Figure 60 shows the online performance of sim-DP using Gaussian kernel-based averager. The sim-DP method showed significant improvement over the starting suboptimal PI controllers, but was not as good as the optimal ∞-horizon MPC, or sim-DP with kNN. This is because the Gaussian kernel-based averager assigns a high cost J_{max} if there are less than N_{min} points within a hyper-ellipse of radius ρ. The optimal controller visits regions of the state space not previously visited by the suboptimal schemes. Hence the controller avoided this region during online control. In contrast, kNN approximates the cost-to-go of a query point as a weighted average of cost-to-go of its k nearest neighbors. The relative distances of these neighbors are important, not the actual distances from the query point. Hence, the controller using kNN give nearly optimal performance. We then modified the kNN as:

$$\tilde{J}(x_q) = \begin{cases} \dfrac{\sum_{i=1}^{l} \gamma_i J(\hat{x}_i)}{\sum_{i=1}^{k} \gamma_i} & \text{if } N > k \\ J_{max} & \text{otherwise} \end{cases} \tag{93}$$

where N is the number of data points within hyper-ellipse of radius ρ, k is the number of nearest neighbors, and γ_i are defined in (69). With this modification, sim-DP with kNN performed similar to the Gaussian kernel-based example.

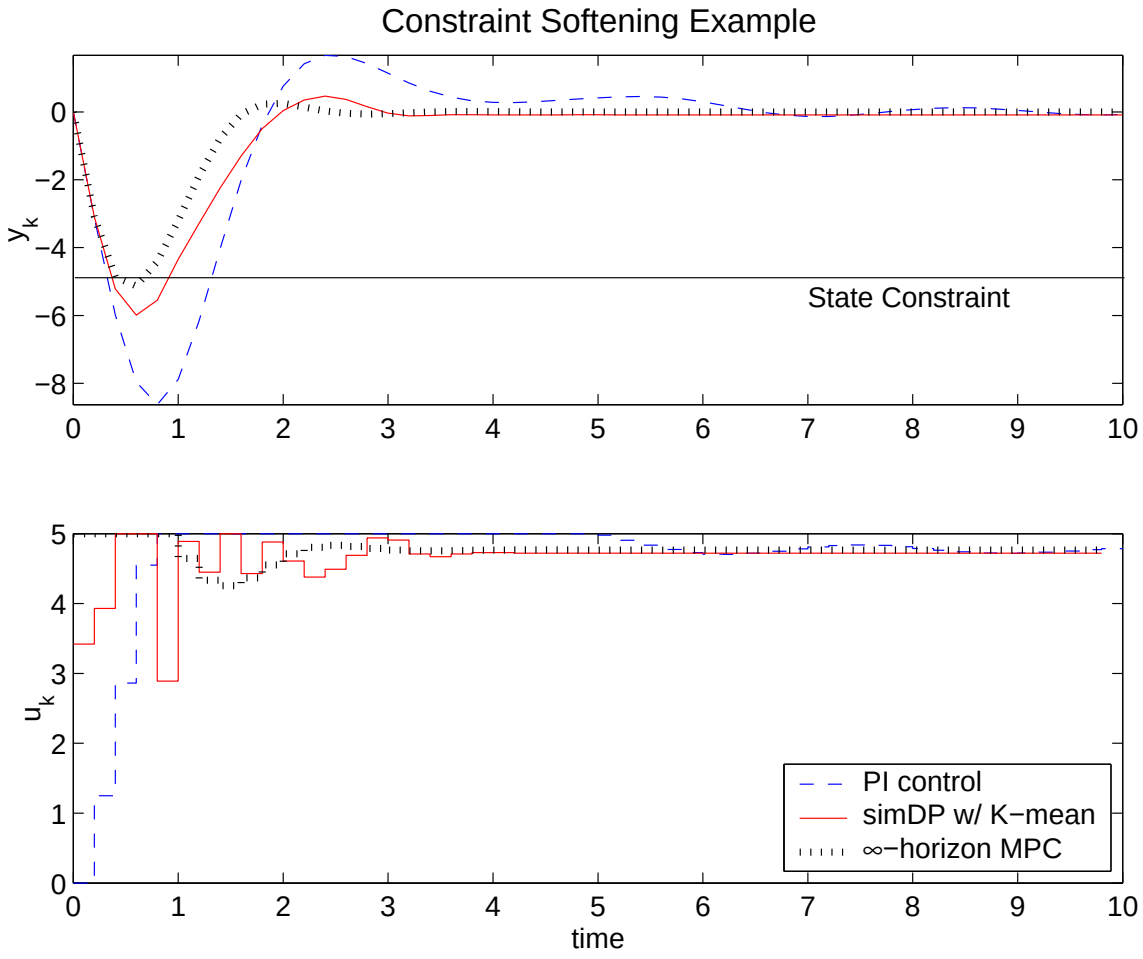

Figure 60: Online performance of value iteration (—), original PI controller (– –) and ∞-horizon MPC ($\cdots$).

Table 18: Comparison of value and policy iteration using neural network and Gaussian kernel-based averager for the nonlinear bioreactor.

Approximator	Gaussian kernel			neural network		
Algorithm	VI	iter-PI	λ-PI ($\lambda = 0.5$)	VI	VI-Restrict	iter-PI
Offline cost-to-go calculation						
Iterations	33	3	14	4	4	7
Pol Evaluation	—	$*$	58	—	—	16
States Added	— Not Applicable —			0	0	245
Comput. Time (hr)	22.9	$> 60^*$	11.0	— Not Applicable† —		
Online performance						
Total Cost $J(x_0)$	9.9	9.9	9.9	24.2	9.4	9.1
Sim. Time (s)	141.9	134.1	134.1	98.7	127.7	69.3

VI-Restrict: Restricting the neural network to visited subset

* Truncated after 40,000 evaluations

† Computation time for neural network wasn't computed

This highlights the importance of data coverage and appropriate approximator choice. If adequate data is not available in the region that the optimal controller visits, the performance of sim-DP will not be optimal. As the true optimal controller is unknown in most problems, a robust exploration scheme would be necessary to improve the controller performance.

7.4.3 Nonlinear bioreactor

We revisit the problem of steady state switching in a continuous bioreactor presented in the previous chapter. The control objective is the optimal switching from the low to high biomass yield steady state. The results are summarized in Table 18 and are discussed here.

7.4.3.1 Convergence of policy iteration

We first compare the offline convergence behavior of value and policy iteration with Gaussian kernel-based averager. The top half of Table 18 shows the comparison of offline convergence

results. Value iteration converged in 33 iteration, taking about 23 hours. On the other hand, for the first iteration of iter-PI, policy evaluation did not converge even after 2 days of simulation and 40,000 iterations. At this stage, we truncated policy evaluation and used the cost-to-go approximator available at this stage to continue policy iterations. We found that only two more iterations (with 18 more policy evaluations) were required for convergence. Next, we considered λ-PI with $\lambda = 0.5$, which converged in 14 iteration requiring 11 hours of offline computation time. All the three approximators yielded optimal performance when used for online control with comparable online simulation times.

7.4.3.2 Exploration of the state space

In previous chapter, we observed that value iteration using a neural network was sensitive to over extrapolation in the region with no data, resulting in a poor performance. By restricting the search to the visited subset of the state space, over extrapolation problem of neural network was avoided, and the control performance improved. Gaussian kernel-based sim-DP controller, on the other hand, showed optimal performance due to its non-expansion properties. Figure 61 shows the state space plot for neural network-based (diamonds) and Gaussian kernel-based control, with the original data shown as dots.

We also incorporated guarded exploration of the state space within the neural network-based policy iteration. Since the approximator is valid only in the visited subset of the state space, it cannot be reliable used to obtain $J\left(f_h(x, \mu^{i+1}(x))\right)$ if an improved policy visits region with inadequate data during offline learning. In such a case, we perform simulations of the policy $\mu^{i+1}(x)$ until the set point is reached. The cost-to-go is then calculated as in sim-PI and all the "new" points visited added to the original data. Figure 62 shows that the policy iteration with neural network sufficiently explore state space beyond the visited region. Crosses ($\times$) represent the additional 245 data points added by exploration. Increased coverage improved the cost-to-go approximations provided by neural network, resulting in optimal online performance. Recall that this result is similar to the "policy update" modification of the original value iteration, where single sweeps of policy iteration were used within each value iteration for each state-action pair that visited regions of state

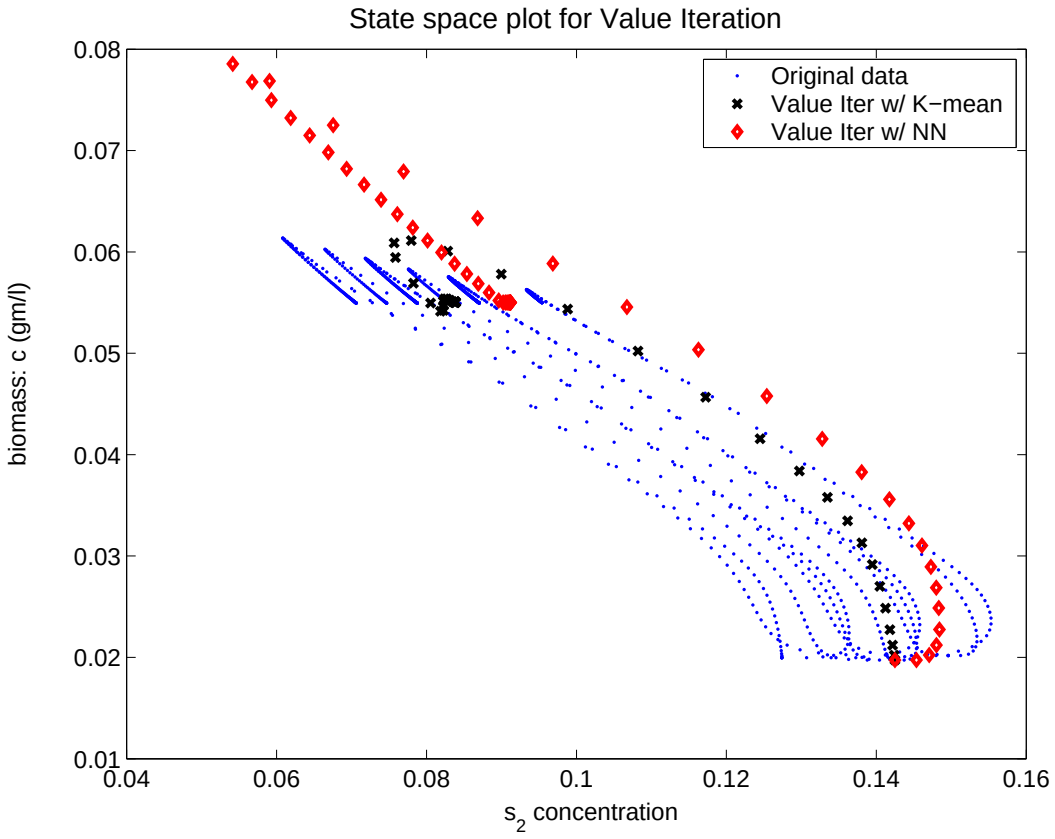

Figure 61: State space plot (c vs s_2) showing original MPC data points (dots) and points visited during online control using cost-to-go approximator from value iteration using i. Neural Network ($\diamondsuit$) and ii. K-mean clustering ($\times$)

space with insufficient data.

7.4.4 Further Examples

Figure 62 shows that the optimal trajectory for the bioreactor steady state switching problem lies within the visited subset of the state space. As a result, even though exploration of the state space improves control performance, conservative schemes with safeguards against extrapolation work equally well. However, this will not always be the case. One is likely to encounter situation like the one shown in Figure 63, where the optimal trajectory lies beyond the visited subset of state space. In such a case, guarded exploration during policy iteration is likely to produce significantly better performance than value iteration restricted to the visited subset.

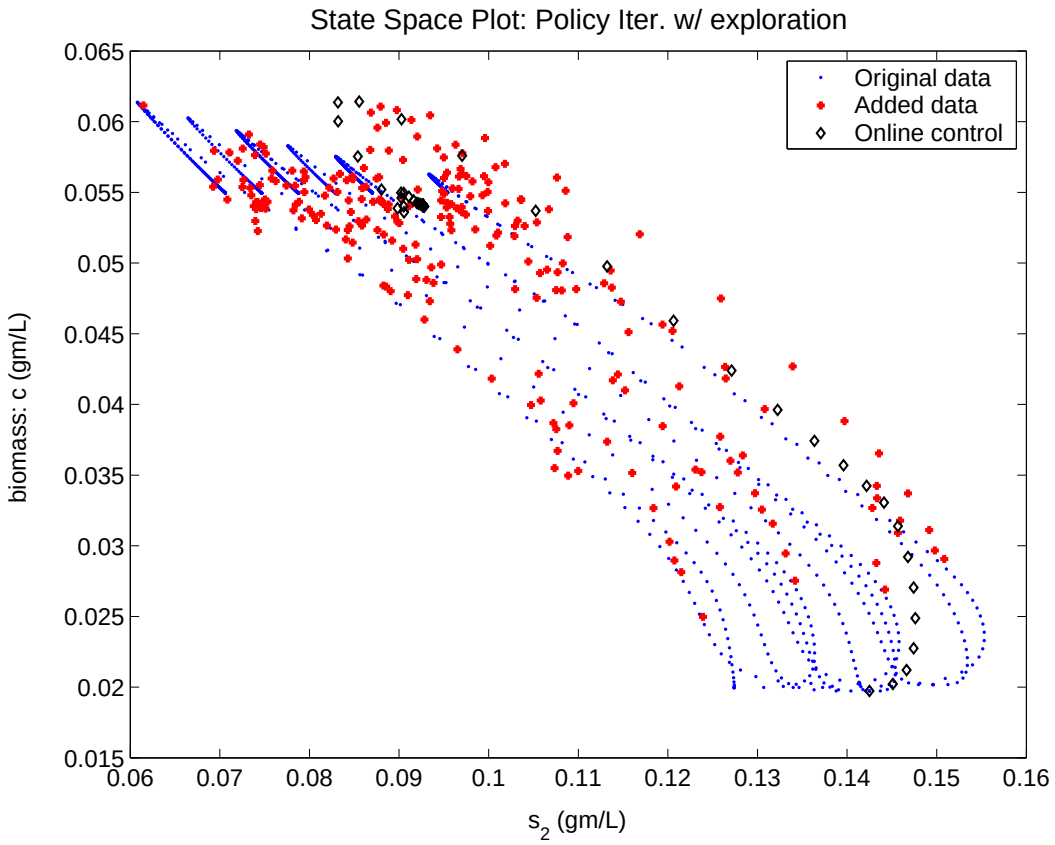

Figure 62: State space plot (c vs s_2) for policy iteration using neural network. Data points added during "policy update" ($\times$) increases the coverage of state space and results in optimal control performance ($\diamondsuit$)

7.5 Conclusions

In this paper, we first compared global parametric approximator such as neural network with local averager for approximating the cost-to-go function. The local averagers — k-nearest neighbor and Gaussian kernel-based averager — provided stable and monotonic learning of the cost-to-go function and more robust online performance than the global approximator.

The speed of convergence of value and policy iteration were compared next. Through an analytical linear quadratic control example, as well as constrained linear / nonlinear systems example, we showed that policy iteration takes lesser number of iteration, but requires more function evaluations in the policy evaluation step. Two alternate ways to perform policy evaluation — based on iterative evaluation, and simulation-based calculation — were presented. We also showed that the λ-policy iteration scheme of Bertsekas and Ioffe

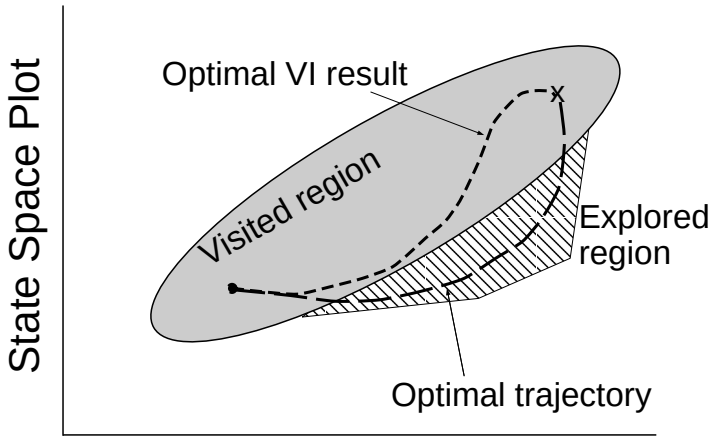

Figure 63: Cartoon depicting benefits of exploration.

[11] effectively uses value iteration to reduce the number of policy evaluations.

Finally, we discussed the ability of policy iteration to perform guarded exploration in the unvisited subset of the state space, can lead to more optimal results. This exploration can be guided by extrapolation of the function approximator, or by using dithering input signal during offline learning.

CHAPTER VIII

CONTRIBUTIONS AND FUTURE WORK

8.1 Summary of Contributions

The two vignettes of switching systems, viz. the switching of the flow direction in a microreactor and optimal switching between metabolic states of a bioreactor were considered in this thesis. An analysis of the reverse-flow operation and its comparison with the unidirectional operation of the microreactor was presented in the first part of the thesis. The part-II of the thesis looked into the implementation of a simulation-based Approximate Dynamic Programming (ADP) approach for optimal steady state switching. The *major* contributions of this thesis are:

- Developing a physical understanding of the observed improvement in the performance of the reverse-flow microreactor over a unidirectional one. A simple scaling analysis was employed to demonstrate that the thermal dynamic properties of the reactor were critical in determining the optimal reverse-flow operation. Parametric study was undertaken to study the effect of various input parameters, system properties and reaction / transport parameters, and to define the performance limits of the reactor operation.

- Understanding the issues of coverage of the state space, the choice of the function approximator and the choice of the iteration algorithm in implementation of simulation-based Approximate Dynamic Programming for optimal switching of steady states in a bioreactor. It was shown that avoiding extrapolation beyond the visited subset (the "training set") of the state space and / or exploration within the unvisited subset are critical in obtaining a good control performance.

The specific achievements are summarized in the remainder of this section.

Part-I of this thesis was motivated by the experimental observation that the reverse-flow (RF) operation of a micro-channel reactor provided better hydrogen yield than the unidirectional (UD) operation [59]. The aim of this work was to investigate the physical origins of this observation, and to obtain operating and design guidelines for methane partial oxidation in a microreactor. The overreaching objective was to arrive at a general framework, which would be system independent, though the specific results pertained to the experimental example.

In chapter 2, a 1-D mathematical model of methane partial oxidation in a microreactor was presented. Reaction kinetics rate equations from various sources were critically compared and the experimentally observed improvement in hydrogen yield was reproduced through simple kinetic equations. An analysis of time scales of various processes operating within the reactor was presented in order to gain more insight into the reactor operation. Radiation heat transfer, which has largely been neglected in the partial oxidation / reforming literature, in spite of high operating temperatures, was considered. The radiation model demonstrated that although the radiation flux is substantial, the large aspect ratio ($l/d = 232$) of the channels precludes any significant radiation effects. This effect however becomes more significant in a shorter reactor with smaller aspect ratio ($l/d = 100$).

The microreactor model so developed was then used for a detailed analysis of the microreactor operation in Chapter 2 and for operability and sensitivity analysis in Chapter 4. The effect of various input parameters such as the inlet feed ratio, input velocity, and inlet temperature, as well as that of the switching time was investigated. The three key advantages of the RF operation — viz. favorable reaction thermodynamics, better thermal utilization, and exploitation of regenerative heat exchange — were shown to provide process improvement in the microreactor under various operating conditions. The optimal switching of the inlet-outlet ports was shown to occur near the time scale of the reaction heat release. This thesis demonstrated that the main advantage of the reverse-flow operation is its robustness to various changes in the kinetic and transport parameters in the system, which is a critical consideration for any practical system operated under real-life conditions. For higher hydrogen throughput, the reverse-flow operation is required as the

unidirectional autothermal state quenches at higher velocities. A rational means of improved catalyst placement that exploits the spatial temperature patterns in the reverse-flow microreactor was also presented. An opposed flow reactor that retains some of the advantages of the reverse-flow operation, without requiring flow reversal or moving reactor parts, was demonstrated.

Part-II of this thesis considered a novel simulation-based Approximate Dynamic Programming (ADP) framework for optimal control of switching between multiple steady states in a bioreactor. The cybernetic modeling framework of Ramkrishna and coworkers was used for modeling continuous microbial [62] and mammalian [79] reactors. A sequential linearization based MPC algorithm was used for optimal switching between multiple steady states in both the systems. While MPC was able to drive the reactors to the desired steady state from the undesirable one, it suffered from two drawbacks: high computational load and suboptimal performance due to the short horizon size. Therefore, the ADP method, motivated by work in AI field of Reinforcement Learning, was developed [69] and applied [58] to the optimal steady state switching problem in the microbial bioreactor.

Chapter 6 presented the results of applying the ADP method to the optimal steady state switching problem. The computational burden of solving an infinite horizon control problem is avoided through the use of "value" or "cost-to-go" function, which is obtained as a function of the system state offline using a learning algorithm on a restricted subset of the state space defined by simulations. The online control problem is reduced to an equivalent single stage problem. As the function approximator is obtained over this subset of state space, guarding against extrapolation is required to ensure good online control performance. Chapter 6 also presents a way for judicious exploration of the unvisited regions of the state space.

Finally, Chapter 7 investigated the key issues in implementation of ADP through various linear and nonlinear examples: viz. choice of function approximator, learning algorithm used and exploration of the state space during offline learning. A local averager gave superior performance than a global parametric approximator (see [67] for details). The convergence properties and exploration abilities of value iteration and policy iteration algorithms were

also discussed.

8.2 Future Work

The possible avenues for extension of the work on a reverse-flow microreactor include:

- **Experimental verification**

  The initial results presented in Chapter 2 reproduced the experimental observations of Kikas et al. [59]. The experimental verification of other results pertaining to the optimal feed conditions, optimal switching time and the operating diagrams (Fig. 33) would greatly increase their fidelity. The catalyst placement results and the effects of changing the reactor material of construction from Chapter 4 will benefit greatly if backed by experimental data. With advances in Micro Electro Mechanical Systems (MEMS) technology, an experimental reactor based on planar design shown in Fig. 38 can also be built and tested.

- **Development of multidimensional CFD models**

  The straightforward extension of this work includes considering a full-scale 3-D transient CFD (computational fluid dynamics) model for the reactor. While a 3-D model may not necessarily provide greater insight into the basic physics of the problem (as compared to 1-D simulations or scale analysis), it allows to study the spatial effects and is useful in analyzing the interaction of transport and reaction processes for the complex reactor geometries of the flow channels. It would also allow to study the effect of reactant mal-distribution, as well as different conditions seen by the channels inside the reactor core as compared to those near the periphery.

  Finally, the assumption of diffuse-grey emission of radiation in a single channel may also be relaxed. Radiation effects considering interaction between different channels can also be considered [106].

- **Development of multi-scale models**

  Since the system is kinetically limited, incorporating detailed microkinetic models [47, 27] is likely to provide more accurate results. The microkinetic models for surface

reactions can be combined with gas phase reaction chemistry and flow simulations. A hierarchical multi-scale simulation approach [89] can be applied to this effect.

- **Improved reactor design**

  An integrated theoretical-experimental approach will greatly help in improved design of a microreactor with complex flow geometries and spatial patterning of catalytic, sensor or actuator elements. Due to the significant advances in MEMS, it is now possible to build such systems. In this regard, it is expedient and economical to employ detailed models to analyze various different spatial flow / element patterns for selecting an optimal reactor design [110].

- **Optimization and control**

  Optimization of periodic systems is still an open issue. As the system dynamics have a fast response, an online optimization scheme is not suitable. However, newton-picard algorithm can be employed to obtain quick periodic steady state solutions and employ them within an optimization framework. Recently, there has been some work on control of reverse-flow reactors. For example, Dufour et al. [28] considered an MPC-IMC (Internal Model Control) scheme for control of a reverse-flow catalytic oxidation process, while Fissore et al. [35] developed a control-relevant neural network model for their "ring" reactor. Instead of the more computationally cumbersome MPC-IMC scheme of Dufour, a repetitive MPC scheme [64] can be implemented instead using a linearized model or a neural network-based NARX (nonlinear auto regressive with exogenous input) model. The next step would be to use the approximate Dynamic Programming in a hierarchical control scheme [68].

There are two directions for continuing the work on bioreactor control: algorithmic approach and systematic approach. The former one involves development of algorithms, using the bioreactor system as a "test bed," while the latter one treats the specific class of bioreactors as a common system for generating "tailored" optimization and control solutions. The several avenues for extension of the work on bioreactor are as follows:

- **Cybernetic Model based Control**

  One of the weak points of the work presented in Part-II is that a consistent algorithmic or systematic approach was not employed. Chapter 5 was devoted to obtaining problem-specific MPC solutions for the two systems; Chapters 6 and 7 focused on the ADP algorithms to solve the highly nonlinear optimal control problem. With some maturity reached in ADP methods [66], a systematic approach for developing model-based control techniques for control of bioreactors described by the cybernetic model framework can be undertaken. The hybridoma system can be used as an example and techniques geared at addressing specific issues (see Chapter 5) common to the cybernetic modeling framework can be developed. Specifically, a multiple model based MPC guided by a supervisory ADP scheme [66, 68] can be employed.

- **Optimal Control of Batch Reactors and Reactor Startup**

  Optimal control of batch and fedbatch reactors has mainly focussed on obtaining open-loop optimal solutions. The optimal open loop trajectory is obtained offline, and sometimes a shrinking horizon control is used to track this trajectory [99]. An alternative is to use the ADP approach, as was shown in our previous paper [86], to obtain a closed-loop optimal policy, expressed as a function of system state. In case of a fixed end-time problem, one of the issues is that the Bellman equation does not have a fixed point solution, but a fixed trajectory solution; i.e. individual cost-to-go functions at each sample time in the horizon. Judicious exploration of state space and choice of learning algorithm are important properties that can be investigated. Finally, the possibility of learning the optimal cost-to-go during online implementation of the policy (on-policy sim-DP) is another issue to be investigated.

- **Policy Gradient Scheme**

  The sim-DP schemes investigated so far aimed at obtaining the cost-to-go function approximator. An alternative to encoding the cost function is to directly express the optimal policy as a function of system state. A parametric function is used to approximate the current policy $\tilde{\mu^i}(x)$, and the Bellman iterations are used to determine

the new policy based on the gradient of the current policy. Hence the name policy gradient. While this method has been investigated in the artificial intelligence community [118, 105], several modifications need to be made to adapt it to the control problems. Translation of this method to continuous state / action spaces and selection of appropriate approximation schemes are the specific issues to be addressed.

APPENDIX A

FINITE AREA VIEW FACTORS

Consider two ring elements at locations m and n as shown in Figure 64. Let z_m represent the distance from the left end of the channel to the right boundary of m^{th} grid and Δz_m represent the axial extent of the grid element (*i.e.* $\Delta z_m = z_m - z_{m-1}$). Let p, q, r and s represent the cylindrical surfaces and 0, $m-1$, m, $n-1$ and n represent the faces.

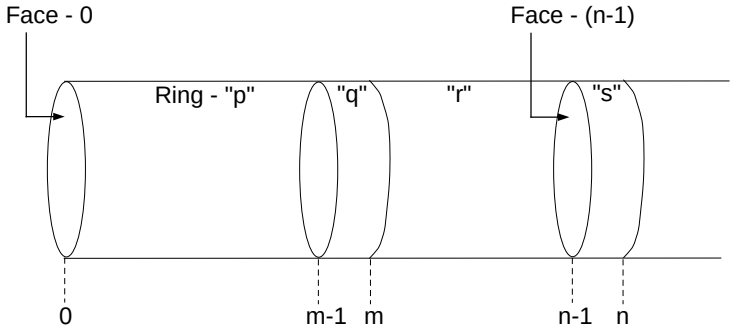

Figure 64: Surfaces considered for view factor calculation.

The dimensionless lengths are defined as

$$X_n = \frac{z_n}{d}, \qquad \Delta X_n = \frac{\Delta z_n}{d}$$

The cross sectional areas are $A_0 = A_m = A_n = \pi d^2/4$ and the ring surface area is $A_s = \pi d \Delta z_n$. Hence, the ratio of areas is

$$\frac{A_0}{A_s} = \frac{1}{\Delta X_n} \tag{94}$$

The view factors satisfy the following equations

$$\text{For an enclosure } \sum_j F_{i-j} = 1; \qquad A_i F_{i-j} = A_j F_{j-i}$$

The above equations are known as the "summation rule" and the "reciprocity rule" respectively [77].

Finally, the view factor between two parallel coaxial disks of radii r separated by a distance z is given by [77]:

$$G = 0.5\left(\xi - \sqrt{\xi^2 - 4}\right) \tag{95}$$

where G represents the end-end view factor, and $\xi = 2 + (z/r)^2 = 2 + 4X^2$. Thus,

$$G_{i-j} = 1 + 2(X_i - X_j)^2 - 2\sqrt{(X_i - X_j)^4 + (X_i - X_j)^2} \tag{96}$$

A.1 Ring-End View Factor

We seek to obtain $F_{re}(z_n) \triangleq F_{s-0}$. We note the reciprocity rule:

$$F_{s-0} = \frac{A_0}{A_s} F_{0-s} \tag{97}$$

The summation rules for cylindrical enclosures from z_0 to z_n, and that from z_0 to z_{n-1} are given as:

$$\begin{aligned} F_{0-(pqr)} + F_{0-s} + G_{0-n} &= 1 \\ F_{0-(pqr)} + G_{0-(n-1)} &= 1 \end{aligned} \tag{98}$$

$$F_{0-s} = G_{0-(n-1)} - G_{0-n} \tag{99}$$

Using the reciprocity rule (97), we obtain

$$F_{re}(z_n) = \frac{1}{2\Delta X_n}\left[\sqrt{X_n^4 + X_n^2} - \sqrt{X_{n-1}^4 + X_{n-1}^2} + X_{n-1}^2 - X_n^2\right] \tag{100}$$

A.2 View Factor from Ring to Itself

To find the view factor for the n^{th} ring to itself, we use the summation rule for the enclosure formed by the ring s and its faces $n-1$ and n:

$$F_{s-(n-1)} + F_{s-s} + F_{s-n} = 1 \tag{101}$$

We note that due to the symmetry, $F_{s-(n-1)} = F_{s-n}$. They represent the view factor from a ring element to one of its faces. Equation (100) represents the view factor from the ring to face located at $z = 0$. The view factor from the ring to face located at z_{n-1}

is just $F_{re}\left(|z_n - z_{n-1}|\right)$ (ie. in (100), replace X_n with ΔX_n and X_{n-1} with 0). Through straightforward manipulations, one can verify that

$$F_{n-n} \triangleq F_{s-s} \quad = \quad 1 + \Delta X_n - \sqrt{\Delta X_n^2 + 1} \tag{102}$$

A.3 Ring-Ring View Factor

To obtain the ring-ring view factor $F_{n-m} \triangleq F_{s-q}$, we apply the summation rule to the cylindrical enclosures from z_{m-1} to z_n, and from z_m to z_n

$$F_{s-n} + F_{s-s} + F_{s-r} + F_{s-q} + F_{s-(m-1)} \quad = \quad 1 \tag{103}$$

$$F_{s-n} + F_{s-s} + F_{s-r} + F_{s-m} \quad = \quad 1 \tag{104}$$

Subtracting the two and rearranging, we get

$$F_{s-q} = F_{re}(|z_n - z_m|) - F_{re}(|z_n - z_{m-1}|) \tag{105}$$

From Eq. (100), for $i = m-1, m$

$$F_{re}(|z_n - z_i|) = \frac{1}{2\Delta X_n}\left[(X_{n-1} - X_i)^2 - (X_n - X_i)^2 + \Xi_{n-i} - \Xi_{(n-1)-i}\right] \tag{106}$$

$$\text{where} \quad \Xi_{i-j} \triangleq \sqrt{(X_i - X_j)^4 + (X_i - X_j)^2} \tag{107}$$

After straighforward manipulations, we obtain the ring-ring view factor:

$$F_{n-m} = \frac{1}{2\Delta X_n}\left[2\Delta X_m \Delta X_n + \Xi_{n-m} + \Xi_{(n-1)-(m-1)} - \Xi_{(n-1)-m} - \Xi_{n-(m-1)}\right] \tag{108}$$

APPENDIX B

SIM-DP ALGORITHMS

Following notations are used in this thesis. $J(x)$ represents cost-to-go values for a state x; J^* represents the optimal cost-to-go; $\tilde{J}$ represents function approximation of cost-to-go; and $\mu(x)$ represents control policy. Superscript $()^i$ represents iteration index. J^μ represents cost-to-go obtained on following a policy $\mu(x)$. $J(k)$ and $\phi(k)$ are shorthand representations of $J(x(k))$ and $\phi(x(k), u(k))$ respectively, where k is the time index.

B.1 Value Iteration

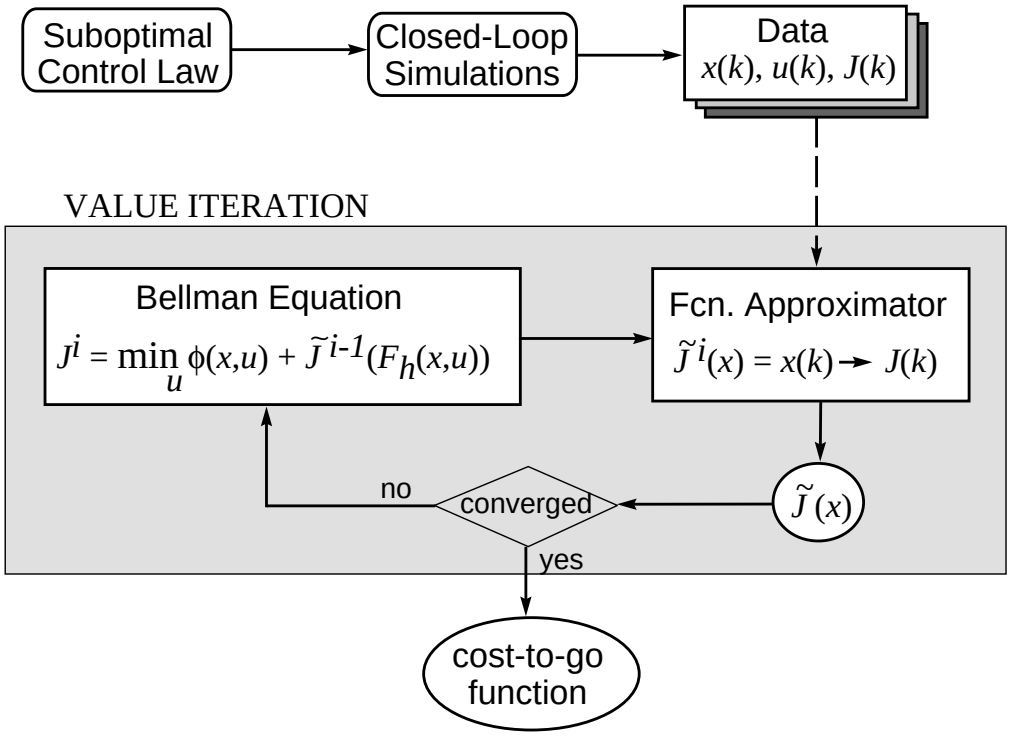

Figure 65: Architecture for offline computation of cost-to-go approximation using value iteration

Figure 65 demonstrates the value iteration algorithm. Various steps in implementation

of the algorithm are

- Perform simulations of the process with chosen suboptimal policies for a set of representative conditions. The states x_k visited by the suboptimal policies $\mu_k^0(x_k)$ form the "visited" region of the state space $\mathcal{X}^{sim}$.

- Evaluate the cost-to-go $J^0(x_k)$ for each $x_k \in \mathcal{X}^{sim}$:

$$J^0(x_k) \triangleq \sum_{j=k+1}^{N} \phi\left(x_j, \mu_j^0(x_j)\right)$$

  The cost-to-go is the sum of single-stage costs until the end of horizon. N is chosen sufficiently large for the system to reach equilibrium, so that $J^0(x_k)$ calculated above is the ∞-horizon cost-to-go for the state x_k.

- Train the neural network or the cost-to-go approximator of choice; represent the approximate cost-to-go as a function $\tilde{J}^0(x_k)$ of the state x_k.

- Perform the following iteration:

  - For each $x_k \in \mathcal{X}^{sim}$, calculate the new cost-to-go as:

$$J^{i+1}(x_k) = \min_{u_k} \left\{\phi(x_k, u_k) + \tilde{J}^i\left(f_h(x_k, u_k)\right)\right\}$$

    which is based on the Bellman equation 62.

  - Fit a new cost-to-go approximator $\tilde{J}^{i+1}$ to the x_k vs. $J^{i+1}(x_k)$ data

  - Increment i; calculate $\delta = \frac{\|J^{i+1} - J^i\|}{\|J^i\|}$

  until convergence, *i.e.* $\delta < tol$

In case of a local averager such as k-nearest neighbor method, the state-cost information is retained in the memory and the training of cost approximation is not required. Cost-to-go values are computed by averaging over the cost-to-go values of neighboring states in the memory.

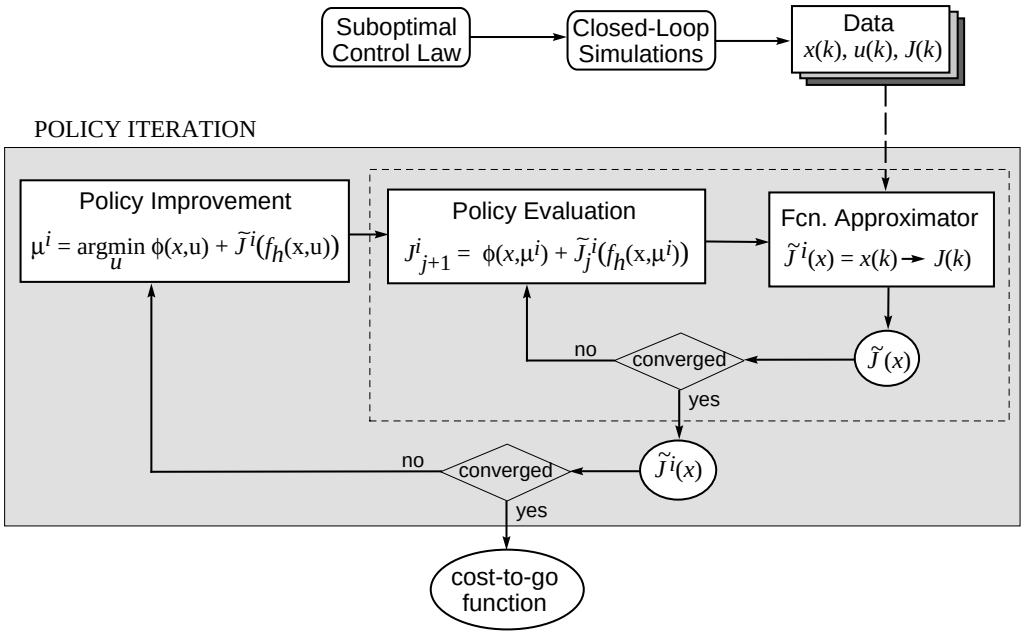

Figure 66: Architecture for offline computation of cost-to-go approximation using policy iteration

B.2 Policy Iteration

Figure 66 demonstrates the policy iteration algorithm. This algorithm consists of two distinct steps: policy improvement step and an iterative policy evaluation step. Various steps in implementation of the algorithm are

- Initialize $\tilde{J}^0(x_k)$ and $\mathcal{X}^{sim}$ using suboptimal policies $\mu_k^0(x_k)$ as shown in the first three steps in the previous algorithm.
- Perform policy improvement for each $x_k \in \mathcal{X}^{sim}$

$$\mu^{i+1}(x_k) = \arg\min_{u_k} \left\{ \phi(x_k, u_k) + \tilde{J}^i \left(f_h(x_k, u_k) \right) \right\}$$

- Initialize the cost-to-go approximator for policy evaluation as $\tilde{J}^{i+1,0} = \tilde{J}^i$
- Perform the policy evaluation by repeating
  - If $f_h\left(x_k, \mu^{i+1}(x_k)\right) \in \mathcal{X}^{sim}$, then calculate the new cost-to-go

$$J^{i+1,j+1}(x_k) = \phi\left(x_k, \mu^{i+1}(x_k)\right) + \tilde{J}^{i+1,j}\left(f_h\left(x_k, \mu^{i+1}(x_k)\right)\right)$$

- Fit a new cost-to-go approximator $\tilde{J}^{i+1,j+1}$ to the x_k vs. $J^{i+1}(x_k)$ data
- Increment j; calculate $\delta = \frac{\|J^{i+1,j+1} - J^{i+1,j}\|}{\|J^{i+1,j}\|}$

until convergence, *i.e.* $\delta < tol$. We represent this cost-to-go as $\tilde{J}^{i+1}$.

- Terminate the policy iteration if $\mu^{i+1} \approx \mu^i$. Otherwise, go to the second step and perform policy improvement and policy evaluation again.

Remark 1 *If $f_h\left(x_k, \mu^{i+1}(x_k)\right) \notin \mathcal{X}^{sim}$, we use additional simulations to perform policy evaluation and update $\mathcal{X}^{sim}$ accordingly. We call this method "policy update" and is discussed in section 7.2.3.*

Remark 2 *The convergence criterion $\mu^{i+1} \approx \mu^i$ can be equivalently stated as $\frac{\|J^{i+1} - J^i\|}{\|J^i\|} < tol$. While this isn't completely accurate, its a acceptable for our examples.*

APPENDIX C

PROOF OF (87)

For the linear system, we have

$$\phi(x_{k+m}) = x_k^T \left[A - BL^{i+1}\right]^{mT} \left[Q + L^{(i+1)T} R L^{i+1}\right] \left[A - BL^{i+1}\right]^m x_k \quad (109)$$

$$J(x_{k+m}) = x_k^T \left[A - BL^{i+1}\right]^{mT} S^i \left[A - BL^{i+1}\right]^m x_k \quad (110)$$

We will drop the superscript from L^{i+1} and S^i for notational convenience, but any other superscript will be denoted explicitly. As shown in Eq. (83), we will write $[A - BL] = Y$ and $[Q + L^T RL] = Z$. For the linear system, TD becomes

$$d(x_{k+m}) = x_k^T \left\{Y^{mT} Z Y^m + Y^{(m+1)T} S Y^{m+1} - [A - BL]^{mT} S [A - BL]^m\right\} x_k \quad (111)$$

Using (111), we can verify that

$$\sum_{m=0}^{\infty} \lambda^m d(x_{k+m}) = x_k^T \left\{ \sum_{m=0}^{\infty} \lambda^m \left[Y^{mT} Z Y^m + Y^{(m+1)T} S Y^{m+1}\right] - S - \sum_{m=1}^{\infty} \lambda^m [A - BL]^{mT} S [A - BL]^m \right\} x_k \quad (112)$$

We note that $\sum_{m=0}^{\infty} \lambda^m d(x_{k+m}) + x_k^T S x_k$ is the new cost-to-go $x_k^T S^{i+1} x_k$. In the last summation in (112), we replace $m = l + 1$ to obtain

$$\begin{aligned} S^{i+1} &= \sum_{m=0}^{\infty} \lambda^m \left[Y^{mT} Z Y^m + Y^{(m+1)T} S Y^{m+1}\right] - \lambda \sum_{l=0}^{\infty} \lambda^l [A - BL]^{(l+1)T} S [A - BL]^{l+1} \\ &= \sum_{m=0}^{\infty} \lambda^m \left[Y^{mT} Z Y^m + (1 - \lambda) Y^{(m+1)T} S Y^{m+1}\right] \quad (113) \\ &= \sum_{m=0}^{\infty} [\lambda^{0.5} Y]^{mT} \left[\lambda Z + (1 - \lambda)[Z + Y^T S Y]\right] [\lambda^{0.5} Y]^m \end{aligned}$$

Through straightforward algebra and using Eq. (79), we can show that

$$[Z + Y^T S Y] = A^T S A + Q - A^T S B [B^T S B + R]^{-1} B^T S A \triangleq S^{i,rd} \quad (114)$$

We write $\tilde{Y} = \lambda^{0.5}[A - BL]$ and $\tilde{Z} = \lambda[Q + L^T RL] + (1 - \lambda) S^{i,rd}$, and the proof is complete

APPENDIX D

REACTION KINETICS FOR VARIOUS CATALYSTS

The kinetic rate expressions from Gosiewski et al. [43] that were used for our microreactor are shown in Table 2. For all the reactions, the rate expressions are of the form

$$r_j = k_{0,j} e^{-E_j/R.T_s} \left[1 - \prod_{i=1}^{n_{sp}} p_i^{\nu_{ij}} / K_{eq,j} \right]$$

Note that in case of irreversible oxidation reaction, $1/K_{eq,1} = 0$. Table 9 shows the values of $k_{0,j}$ and E_j for the three different catalysts for each of the three reactions. Here we describe the procedure used to obtain these values.

Different sources in the literature report widely varying rate constants and activation energies for the same reaction on the same catalyst. The rates of oxidation, reforming and water gas shift reactions from specific sources [120, 114, 117] are shown in Table 19. We assumed that the ratio of arrhenius rate constants, while the difference between activation energies for a single reaction on different catalysts remains the same. Examples follow.

D.1 Oxidation catalyst

Yao [120] reported activation energies and turnover rates for methane oxidation for average condition of 1% O_2 and 0.1% CH_4 at $500°C$ for Pt and $400°C$ for Pd. Based on this, we estimated the arrhenius constant to be 1.822×10^3 for Pt and 4.4×10^3 for Pd. The ratio of the two was 0.414. Assuming that the ratio remains the same, the Arrhenius constant for Pd is given as

$$k_{0,\mathrm{Pd}} = k_{0,\mathrm{Pt}}/0.414 = 55.69\, mol/m^2.sec$$

In case of activation energies, we assumed that the difference in the activation energies remains the same. Yao [120] reported the difference in activation energies for Pt and Pd to be $-17\, KJ/mol$. Thus, activation energy for Cat1 is $83.32\, KJ/mol$.

Pd is known to be a poor reforming catalyst. As reliable kinetic data for reforming using Pd catalyst is not readily available, we assumed the rate of reforming to be 25 times lower and the activation energy to be the same as that of the nominal case. According to Table 19, the ratio of rate constants for water gas shift reactions was 0.25 and the difference in activation energies $-20\,KJ/mol$. Therefore for Cat1, $k_0 = 1.648$ and $E_a = 58.13$.

D.2 Reforming catalyst

While kinetics of methane oxidation have been extensively studied for noble metal catalysts, relatively few studies were carried out using Ni [26]. Recently, Angelidis and Tzitzios [3] reported Pt/alumina was ten-fold more active than Ni/alumina towards methane oxidation at the stoichiometric point. Hence, we assumed the same activation energy and a ten-fold lower rate constant for oxidation on Ni. Rate constants and activation energies for reforming and water gas shift reactions on Ni are shown in Table 19. Keeping the ratio of rate constants (0.08 for reforming and 80 for water gas shift) and the difference of activation energies (0.0 for reforming and $-27\,KJ/mol$ for water gas shift) the same in our system, we obtain the rates reported in Table 9.

D.3 Shift catalyst

Yao [120] reported oxidation kinetics for Pt, Pd and Rh. Therefore, in this study, we assume Ru to have the same activation energy as Rh ($E_{a,\mathrm{Pt}} - E_{a,\mathrm{Rh}} = -12\,KJ/mol$) towards oxidation and half the rate constant of Pt. The resulting kinetic constants are $k_0 = 11.53$ and $E_a = 112,32$. For reforming reaction, the ratio of kinetic constants on Pt and Ru is 0.426 and the difference of activation energies is $-16\,KJ/mol$. For shift reaction, the Ru catalyst is 16 times more active than Pt and has the same activation energy. Using this information, the remaining kinetic constants in Table 9 were derived.

Table 19: Reaction kinetics for oxidation, reforming and shift reactions from the literature. See the text for description.

Catalyst	m	n	E_a	k_{0f}	Turnover
Oxidation [120] $r = k_0 \exp(-E_a/R.T)\, p_{O2}^m p_{CH4}^n$					
Pt	-0.6	1.0	88	?	0.13
Pd	0.1	0.8	71	?	5.4
Reforming $r_f = k_{0f} \exp(-E_a/R.T)\, p_{H2O}^m p_{CH4}^n$					
Pt	1.0	1.0	75	2.0×10^4	13.1
Ru	1.0	1.0	91	4.7×10^4	4.2
Ni	1.0	1.0	102	2.5×10^5	4.1
Water gas shift [117] $r_f = k_{0f} \exp(-E_a/R.T)\, p_{CO}^m p_{H2O}^n$					
Pt	1.0	*	80	1.0×10^6	13.1
Ru	1.0	*	80	1.6×10^7	4.2
Ni	1.0	*	85	8.0×10^7	4.1
Pd	1.0	*	100	4.0×10^6	13.1

* pseudo zero order

E_a in (KJ/mol), and k_0 in $(sec^{-1}.kPa^{-m-n+1})$

p_i in (kPa)

REFERENCES

[1] For example, see http://www.fischer-tropsch.org/ and documents and references therein.

[2] Agarwal, P., Lee, C., Lim, H. C., and Ramkrishna, D., "Theoretical investigations of dynamic behavior of isothermal continuous stirred tank biological reactors," *Chemical Engineering Science*, vol. 37, pp. 453–462, 1982.

[3] Angelidis, T.-N. and Tzitzios, V., *A comparative kinetic study of* CH_4 *oxidation by Ni/*Al_2O_3*, Pt/*Al_2O_3 *and NiO-Pt/*Al_2O_3 *catalysts*, vol. 122. Amsterdam: Elsevier, 1999.

[4] Aris, R. and Amundson, N. R., "An analysis of chemical reactor stability and control," *Chemical Engineering Science*, vol. 7, pp. 121–131, 1958.

[5] Avci, A. K., Trimm, D. L., and Önsan, Z. I., "Heterogeneous reactor modeling for simulation of catalytic oxidation and steam reforming of methane," *Chemical Engineering Science*, vol. 56, pp. 641–649, 2001.

[6] Bailey, J. E., "Optimal periodic processes in the limits of very fast and very slow cycling," *Automatica*, vol. 8, pp. 451–454, 1972.

[7] Bao, X., Iglesia, E., and Xu, Y., eds., *Natural Gas Conversion VII*, Studies in Surface Science and Catalysis, (Dalian, China), Elsevier, June 2004. Also see the proceedings of the previous Natural Gas Conversion Symposia. References are available from [26].

[8] Bellman, R. E., *Dynamic Programming*. New Jersey: Princeton University Press, 1957.

[9] Bemporad, A., Morari, M., Dua, V., and Pistikopoulos, E. N., "The explicit linear quadratic regulator for constrained systems," *Automatica*, vol. 38, pp. 3–20, 2002.

[10] Bertsekas, D. P., *Dynamic Programming and Optimal Control*. Belmont, MA: Athena Scientific, 2nd ed., 2000.

[11] Bertsekas, D. P. and Ioffe, S., "Temporal difference based policy iteration and applications in neuro-dynamic programming," Report LIDS-P-2349, Lab. for Info. and Decision Systems, Massachussetts Institute of Technology, Cambridge, MA, 1996.

[12] Bertsekas, D. P. and Tsitsiklis, J. N., *Neuro-Dynamic Programming*. Belmont, MA: Athena Scientific, 1996.

[13] Blackwell, D., "Discounted dynamic programming," *The Annals of Matheratical Statistics*, vol. 36, pp. 226–235, 1965.

[14] Blanks, R. F., Wittrig, T. S., and Peterson, D. A., "Bidirectional adiabatic synthesis gas generator," *Chemical Engineering Science*, vol. 45, pp. 2407–2413, 1990.

[15] Boehman, A., "Radiation heat transfer in catalytic monoliths," *A.I.Ch.E. Journal*, vol. 44, pp. 2745–2755, 1998.

[16] Boyan, J. A. and Moore, A. W., "Generalization in reinforcement learning: Safely approximating the value function," in *Advances in Neural Information Provessing Systems* (Tesauro, G., Touretzky, D., and Leen, T., eds.), vol. 7, pp. 369–376, Cambridge, MA: MIT Press, 1995.

[17] Bradtke, S. J., "Reinforcement learning applied to linear quadratic regulation," in *Advances in Neural Information Processing Systems* (Hanson, S. J., Cowan, J., and Giles, G. L., eds.), vol. 5, Morgan Kaufmann, 1993.

[18] Brandner, J. J., Emig, G., Liauw, M. A., and Schubert, K., "Fast temperature cycling in microstructure devices," *Chemical Engineering Journal*, vol. 101, pp. 217–224, 2004.

[19] Branicky, M. S., Borkar, V. S., and Mitter, S. K., "A unified framework for hybrid control: Model and optimal control theory," *IEEE Transactiona on Automatic Control*, vol. 43, pp. 31 – 45, 1998.

[20] Brown, P. N., Hindmarsh, A. C., and Petzold, L. R., "Consistent initial condition calculation for differential-algebraic systems," Report UCRL-JC-122175, Lawrence Livermore National Laboratory, 1995.

[21] Bruns, D. D., Bailey, J. E., and Luss, D., "Steady-state multiplicity and stability of enzymatic-reaction systems," *Biotechnology and Bioengineering*, vol. 15, pp. 1131–1145, 1973.

[22] Cuthrell, J. E. and Biegler, L. T., "Simultaneous optimization and solution methods for batch reactor control profiles," *Computers and Chemical Engineering*, vol. 13, pp. 49–62, 1989.

[23] D, . D., Rosynek, M. P., Kharas, K. C. C., and Lunsford, J. H., "Partial oxidation of methane to carbon monoxide and hydrogen over a Ni/Al_2O_3 catalyst," *Journal of Catalysis*, vol. 132, pp. 117–127, 1991.

[24] De Farias, D. P. and Van Roy, B., "On the existence of fixed points for approximate value iteration and temporal-difference learning," *Journal of Optimization Theory and Applications*, vol. 105, pp. 589–608, 2000.

[25] de Groote, A. M., Froment, G. F., and Kobylinski, T., "Synthesis gas production from natural gas in a fixed bed reactor with reversed flow," *The Canadian Journal of Chemical Engineering*, vol. 74, pp. 735–742, 1996.

[26] de Smet, C. R. H., de Croon, M. H. J. M., Berger, R. J., Marin, G. B., and Schouten, J. C., "Design of adiabatic fixed-bed reactors for the partial oxidation of methane to synthesis gas. application to production of methanol and hydrogen-for-fuel-cells," *Chemical Engineering Science*, vol. 56, pp. 4849–4861, 2001.

[27] Deshmukh, S. R., Mhadeshwar, A. B., and Vlachos, D. G., "Microreactor modeling for hydrogen production from ammonia decomposition on ruthenium," *Industrial and Engineering Chemistry Research*, vol. 43, pp. 2986–2999, 2004.

[28] Dufour, P., Couenne, F., and Touré, Y., "Model predictive control of a catalytic reverse flow reactor," *IEEE Transaction on Control System Technology*, vol. 11, pp. 705–714, 2003.

[29] Ehrfeld, W., Hessel, V., and Lowe, H., *Microreactors: New technology for modern chemistry*. Weinheim: Wiley, 2000.

[30] Eigenberger, G. and Nieken, U., "Catalytic combustion with periodic flow reversal," *Chemical Engineering Science*, vol. 43, pp. 2109–2115, 1988.

[31] Eigenberger, G. and Nieken, U., "Catalytic air purification: Challenges and new solutions," *Chemie Ingenieur Technik*, vol. 63, pp. 781–791, 1991.

[32] Europa, A. F., Gambhir, A., Fu, P.-C., and Hu, W.-S., "Multiple steady states with distinct cellular metabolism in continuous culture of mammalian cells," *Biotechnology and Bioengineering*, vol. 67, pp. 25–34, 2000.

[33] Fischer, V. F. and Tropsch, H. *Brennstoff-Chemie*, vol. 3, p. 39, 1923.

[34] Fissore, D., Barresi, A. A., and Baldi, G., "Synthesis gas production in a forced unsteady-state reactor network," *Industrial Engineering Chemistry Research*, vol. 42, pp. 2489–2495, 2003.

[35] Fissore, D., Barresi, A. A., and Manca, D., "Modelling of methanol synthesis in a network of forced unsteady-state ring reactors by artificial neural networks for control purposes," *Chemical Engineering Science*, vol. 59, pp. 4033–4041, 2004.

[36] Follstad, B. D., Balcarel, R. R., Stephanopoulos, G., and Wang, D., "Metabolic flux analysis of hybridoma continuous culture steady state multiplicity," *Biotechnology and Bioengineering*, vol. 63, pp. 675–683, 1999.

[37] Froment, G. F. and Bischoff, K. B., *Chemical Reactor Analysis and Design*. New York: John Wiley and Sons, $2^n d$ ed., 1990.

[38] Fu, X., Viskanta, R., and Gore, J. P., "Combustion and heat transfer interaction in a pore-scale refractory tube burner," *Journal of Thermophysics and Heat Transfer*, vol. 12, pp. 164–171, 1998.

[39] Gadkar, K. G., Doyle III, F. J., Crowley, T., and Varner, J., "Cybernetic model predictive control of continuous bioreactor with cell recycle," *Biotechnology Progress*, vol. 19, pp. 1487–1497, 2003.

[40] Ganley, J. C., Seebauer, E. G., and Masel, R. I., "Porous anodic alumina microreactors for production of hydrogen from alumina," *AIChE Journal*, vol. 50, pp. 829–834, 2004.

[41] Gordon, G. J., *Approximate Solutions to Markov Decision Processes*. PhD thesis, Carnegie Mellon University, Pittsburgh, PA, 1999.

[42] Gosiewski, K., "Mathematical simulations of reactors for catalytic conversion of methane to syngas with forced concentration cycling," *Chemical Engineering and Processing*, vol. 39, pp. 459–469, 2000.

[43] Gosiewski, K., Bartmann, U., Moszczyński, M., and Mleczko, L., "Effect of the intraparticle mass transport limitations on temperature profiles and catalytic performance of the reverse-flow reactor for the partial oxidation of methane to synthesis gas," *Chemical Engineering Science*, vol. 54, pp. 4589–4602, 1999.

[44] Güttinger, T. E., *Multiple steady states in azeotropic and reactive distillation*. PhD thesis, Swiss Federal Institute of Technology (ETH), Zurich, 1998.

[45] Haynes, T. N., Georgakis, C., and Caram, H. S., "The design of reverse flow reactors for catalytic combustion systems," *Chemical Engineering Science*, vol. 50, pp. 401–416, 1995.

[46] Hernández, E. and Arkun, Y., "Control of nonlinear systems using polynomial ARMA models," *AIChE Journal*, vol. 39, pp. 446–460, 1993.

[47] Hickman, D. A. and Schmidt, L. D., "Production of syngas by direct catalytic oxidation of methane," *Science*, vol. 259, pp. 343–346, 1993.

[48] Hirschenhofer, J. H., Stauffer, D. B., Engleman, R. R., and Klett, M. G., *Fuel Cell Handbook*. Reading, PA: Parsons Corporation, for US DOE, fourth ed., 1998. DOE/FETC-99/1076.

[49] Horn, F. J. M., "Periodic countercurrent processes," *Industrial and Engineering Chemistry Design and Development*, vol. 6, pp. 30–36, 1967.

[50] Horn, F. J. M. and Lin, R. C., "Periodic processes: A variational approach," *Industrial and Engineering Chemistry Design and Development*, vol. 6, pp. 21–30, 1967.

[51] Howard, R., *Dynamic programming and markov processes*. Cambridge Massachussets: MIT Press, 1960.

[52] Hu, W.-S., Zhou, W., and Europa, A. F., "Controlling mammalian cell metabolism in bioreactors," *Journal of Microbiology and Biotechnology*, vol. 8, pp. 8–13, 1998.

[53] Jacobsen, E. W. and Skogestad, S., "Multiple steady states and instabilities in distillation: Implications for operation and control," *Industrial Engineering Chemistry Research*, vol. 34, pp. 4395–4405, 1995.

[54] Jensen, K. F., "Microreaction engineering - is small better?," *Chemical Engineering Science*, vol. 56, pp. 293–303, 2001.

[55] Jensen, K. F. and Ray, W. H., "The bifurcation behaviour of tubular reactors," *Chemical Engineering Science*, vol. 37, pp. 199–222, 1982.

[56] Kaisare, N. S., Lee, J. H., and Fedorov, A. G., "Hydrogen generation via methane partial oxidation in a microchannel reactor with flow reversal: Mathematical formulation and scaling," *AIChE Journal*, 2004. submitted.

[57] Kaisare, N. S., Lee, J. H., and Fedorov, A. G., "Hydrogen generation via methane partial oxidation in a microchannel reactor with flow reversal: Simulation and analysis of reverse-flow operation," *AIChE Journal*, 2004. submitted.

[58] Kaisare, N. S., Lee, J. M., and Lee, J. H., "Simulation based strategy for nonlinear optimal control: application to a microbial cell reactor," *Int. J. Robust and Nonlinear Control*, vol. 13, pp. 347–363, 2003.

[59] Kikas, T., Bardenshteyn, I., Williamson, C., Ejimofor, C., Puri, P., and Fedorov, A. G., "Hydrogen production in a reverse-flow autothermal catalytic microreactor: From evidence of performance enhancement to innovative reactor design," *Industrial Engineering Chemistry Research*, vol. 42, pp. 6273–6279, 2003.

[60] Kolios, G., Frauhammer, J., and Eigenberger, G., "Autothermal fixed-bed reactor concepts," *Chemical Engineering Science*, vol. 55, pp. 5945–5967, 2000.

[61] Kolios, G., Gritsch, A., Glöckler, B., Sorescu, G., and Frauhammer, J., "Novel reactor concepts for thermally efficient methane steam reforming: Modeling and simulation," *Industrial and Engineering Chemistry Research*, vol. 43, pp. 4796–4808, 2004.

[62] Kompala, D. S., Ramkrishna, D., Jansen, N. B., and Tsao, G. T., "Investigation of bacterial growth on mixed substrates: Experimental evaluation of cybernetic models," *Biotechnology and Bioengineering*, vol. 28, pp. 1044–1055, 1986.

[63] Lee, J. H. and Cooley, B., "Recent advances in model predictive control," in *Chemical Process Control - V*, pp. 201–216b, 1997.

[64] Lee, J. H., Natarajan, S., and Lee, K. S., "Model based predictive control approach to repetitive control of continuous processes with periodic operations," *Journal of Process Control*, vol. 11, pp. 195–208, 2001.

[65] Lee, J. H. and Ricker, N. L., "Extended kalman filter based nonlinear model predictive control," *Ind. Eng. Chem. Res.*, vol. 33, pp. 1530–1541, 1994.

[66] Lee, J. M., *A Study on Architecture, Algorithms, and Applications of Approximate Dynamic Programming Based Approach to Optimal Control*. PhD thesis, Georgia Institute of Technology, Atlanta, GA, 2004.

[67] Lee, J. M., Kaisare, N. S., and Lee, J. H., "An evolutionary improvement of control policies via approximate dynamic programming: Part 1: A comparative study on the choice of approximator," *Journal of Process Control*, 2004. submitted.

[68] Lee, J. M., Kaisare, N. S., Lee, J. H., and Tosukhowong, T., "Detection and management of abnormal situations at their onset," in *AIChE Annual Meeting*, (Indianapolis, IN), 2002.

[69] Lee, J. M. and Lee, J. H., "Neuro-dynamic programming method for mpc," in *DYCOPS VI*, pp. 157–162, 2001.

[70] Lei, F., Olsson, L., and Jørgensen, S. B., "Experimental investigations of multiple steady states in aerobic continuous cultivations of saccharomyces cerevisiae," *Biotechnology and Bioengineering*, vol. 82, pp. 766–777, 2003.

[71] Ma, L. and Trimm, D. L., "Alternative catalyst bed configurations for the autothermic conversion of methane to hydrogen," *Applied Catalysis A: General*, vol. 138, pp. 265–273, 1996.

[72] Matros, Y. S., *Unsteady Processes in Catalytic Reactors*. Studies in Surface Science and Catalysis, Amsterdam: Elsevier, 1985.

[73] Matros, Y. S., *Catalytic Processes under Unsteady-State Conditions*. Studies in Surface Science and Catalysis, Amsterdam: Elsevier, 1989.

[74] Matros, Y. S., "Forced unsteady-state processes in heterogeneous catalytic reactors," *The Canadian Journal of Chemical Engineering*, vol. 74, pp. 566–579, 1996.

[75] Matros, Y. S. and Bunimovich, G. A., "Reverse-flow operation in fixed bed catalytic reactors," *Catalysis Reviews: Science and Engineering*, vol. 38, pp. 1–68, 1996.

[76] Mayne, D. Q., Rawlings, J. B., Rao, C. V., and Scokaert, P. O. M., "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, pp. 789–814, 2000.

[77] Modest, M. F., *Radiative Heat Transfer*. New York: McGraw-Hill Inc., 1993.

[78] Morari, M. and Lee, J. H., "Model predictive control: past, present and future," *Computers and Chemical Engineering*, vol. 23, pp. 667–682, 1999.

[79] Namjoshi, A. A., Hu, W. S., and Ramkrishna, D., "Unveiling steady-state multiplicity in hybridoma cultures: The cybernetic approach," *Biotechnology and Bioengineering*, vol. 81, pp. 80–91, 2003.

[80] Namjoshi, A. A., Kienle, A., and Ramkrishna, D., "Steady-state multiplicity in bioreactors: Bifurcation analysis of cybernetic models," *Chemical Engineering Science*, vol. 58, pp. 793–800, 2003.

[81] Namjoshi, A. A. and Ramkrishna, D., "Multiplicity and stability of steady states in continuous bioreactors: Dissection of cybernetic models," *Chemical Engineering Science*, vol. 56, pp. 5593–5607, October 2001.

[82] Narang, A., "The steady states of microbial growth on mixtures of substitutable substrates in a chemostat," *Jounal of Theoretical Biology*, vol. 190, pp. 241–261, 1998.

[83] Numaguchi, T. and Kikuchi, K., "Intrinsic kinetics and design simulation in a complex reaction network: steam-methane reforming," *Chemical Engineering Science*, vol. 43, pp. 2295–2301, 1988.

[84] Ormoneit, D. and Sen, S., "Kernel-based reinforcement learning," *Machine Learning*, vol. 49, pp. 161–178, 2002.

[85] Pattekar, A. V. and Kothare, M. V., "A microreactor for hydrogen production in micro-fuel cell applications," *Journal of Microelectromechanical Systems*, vol. 13, pp. 7–18, 2004.

[86] Peroni, C. V., Kaisare, N. S., and Lee, J. H., "Optimal control of a fed batch bioreactor using simulation-based approximate dynamic programming," *IEEE Transactions on Control Systems Technology*, p. submitted, 2003.

[87] Puterman, M. L., *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. New York: John Wiley & Sons, 1994.

[88] Puterman, M. L. and Brumelle, S. L., "On the convergence of policy iteration in stationary dynamic programming," *Mathematics of Operations Research*, vol. 4, pp. 60–69, 1979.

[89] Raimondeau, S. and Vlachos, D. G., "Recent developments on multiscale, hierarchical modeling of chemical reactors," *Chemical Engineering Journal*, vol. 90, pp. 3–23, 2002.

[90] Rankin, A. J., Hayes, R. E., and Kolaczkowski, S. T., "Annular flow in a catalytic monolith reactor: The significance of centerline probe temperatures," *Trans. I. Chem. E., Part A*, vol. 73, pp. 110–121, 1995.

[91] Rostrup-Nielsen, J. R., "Sulfur passivated nickel catalysts for carbon-free steam reforming of methane," *Journal of Catalysis*, vol. 85, pp. 31–43, 1994.

[92] Rouge, A., Spoetzl, B., Gebauer, K., Schenk, R., and Renken, A., "Microchannel reactors for fast periodic operation: The catalytic dehydration of isopropanol," *Chemical Engineering Science*, vol. 56, pp. 1419–1427, 2001.

[93] Santos, M. S. and Rust, J., "Convergence properties of policy iteration," *SIAM Journal of Control and Optimization*, vol. 42, pp. 2094–2115, 2004.

[94] Sapoundjiev, H. and Hayes, R. E., "Unsteady-state processes in catalysis: A special issue of Chemical Engineering Science," *Chemical Engineering Science*, vol. 59, pp. 3981–3982, 2004.

[95] Schouten, J. C., Rebrov, E. V., and de Croon, M. H. J. M., "Miniaturization of heterogeneous catalytic reactors: Prospects for new developments in catalysis and process engineering," *Chimia*, vol. 56, pp. 627–635, 2002.

[96] Sheintuch, M. and Nekhamkina, O., "Comparison of flow-reversal, internal-recirculation and loop reactors," *Chemical Engineering Science*, vol. 59, pp. 4065–4072, 2004.

[97] Silveston, P. L. and Hudgins, R. R., "Periodic temperature forcing of catalytic reactions," *Chemical Engineering Science*, vol. 53, pp. 4043–4053, 2004.

[98] Sneesby, M. G., Tade, M. O., and Smith, T. N., "Steady-state transitions in the reactive distillation of MTBE," *Computers and Chemical Engineering*, vol. 22, pp. 879–892, 1998.

[99] Soni, A. S. and Parker, R. S., "Closed-loop control of fed-batch bioreactors: A shrinking-horizon approach," *Industrial and Engineering Chemistry Research*, vol. 43, pp. 3381–3393, 2004.

[100] Sterman, L. E. and Ydstie, B. E., "The steady-state process with periodic perturbations," *Chemical Engineering Science*, pp. 721–736, 1990.

[101] Sterman, L. E. and Ydstie, B. E., "Periodic forcing of the cstr: An application of the generalized pi criterion," *A.I.Ch.E. Journal*, pp. 986–996, 1991.

[102] Straight, J. V. and Ramkrishna, D., "Cybernetic modeling and regulation of metabolic pathways – growth on complementary nutrients," *Biotechnology Progress*, vol. 10, no. 6, pp. 574–587, 1994.

[103] Stranges, A. N., "The us bureau of mines synthetic liquid fuels programme, 1920-1950s: German connections and american advances," *Annals of Science*, vol. 54, pp. 29–68, 1997.

[104] Sutton, R. S. and Bartow, A. G., *Reinforcement learning: an introduction*. Cambridge Massachussets: MIT Press, 1998.

[105] Sutton, R. S., McAllester, D., Singh, S., and Mansour, Y., "Policy gradient methods for reinforcement learning with function approximation," in *Advances in Neural Information Processing Systems*, vol. 12, (Cambridge, MA), MIT Press, 2000.

[106] Tancrez, M., Hilka, M., and Taine, J., "Comparison between local and homogenized models of radiative transfer in a set of parallel ducts considered as a porous medium," *International Journal of Heat and Mass Transfer*, vol. 45, pp. 173–180, 2002.

[107] Tesauro, G., "Td-gammon, a self teaching backgammon program achieves master-level play," *Neural Computation*, vol. 6, pp. 215–219, 1994.

[108] Trimm, D. L. and Lam, C.-W., "The combustion of methane on platinum-alumina fibre catalysts—i," *Chemical Engineering Science*, vol. 35, pp. 1405–1413, 1980.

[109] Uppal, A., Ray, W. H., and Poore, A., "On the dynamic behaviour of continuous stirred tank reactors," *Chemical Engineering Science*, vol. 29, pp. 967–985, 1974.

[110] van den Berg, A., Olthuis, W., and Bergveld, P., eds., *Micro total analysis systems: μTAS-2000 Symposium Enschede, The Netherlands*, (Boston, MA), Kluwer Academic Publishers, 2000.

[111] Varner, J. and Ramkrishna, D., "Metabolic engineering from a cybernetic perspective - i: Theoretical preliminaries," *Biotechnology Progress*, vol. 15, pp. 407–425, 1999.

[112] Wei, J. and Iglesia, E., "Isotopic and kinetic assessment of the mechanism of methane reforming and decomposition reactions on supported iridium catalysts," *Physical Chemistry Chemical Physics*, vol. 6, pp. 3754–3759, 2004.

[113] Wei, J. and Iglesia, E., "Isotopic and kinetic assessment of the mechanism of reactions of ch4 with co2 or h2o to form synthesis gas and carbon on nickel catalysts," *Journal of Catalysis*, vol. 224, pp. 370–383, 2004.

[114] Wei, J. and Iglesia, E., "Mechanism and site requirements for activation and chemical conversion of methane on supported pt clusters and turnover rate comparison among noble," *Journal of Physical Chemistry B*, vol. 108, pp. 4094–4103, 2004.

[115] Wei, J. and Iglesia, E., "Reaction pathways and site requirements for the activation and chemical conversion of methane on ru-based catalysts," *Journal of Physical Chemistry B*, vol. 108, pp. 7253–7262, 2004.

[116] Wei, J. and Iglesia, E., "Structural requirements and reaction pathways in methane activation and chemical conversion catalyzed by rhodium," *Journal of Catalysis*, vol. 225, pp. 116–127, 2004.

[117] Wheeler, C., Jhalani, A., Klein, E. J., Tummala, S., and Schmidt, L. D., "The water gas shift reaction at short contact times," *Journal of Catalysis*, vol. 223, pp. 191–199, 2004.

[118] Williams, R. J., "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Machine Learning*, vol. 8, pp. 229–256, 1992.

[119] Xu, J. and Froment, G. F., "Methane steam reforming, methanation and water-gas shift: I. intrinsic kinetics," *A.I.Ch.E. Journal*, vol. 35, pp. 88–96, 1989.

[120] Yao, Y.-F. Y., "Oxidation of alkanes over noble metal catalysts," *Industrial and Engineering Chemistry Product Research and Development*, vol. 19, pp. 293–298, 1980.

VITA

Niket Kaisare was born in Mumbai, India. He received his B. Chem. Engg at University of Mumbai, Department of Chemical Technology (UDCT) in 1999. He joined Purdue University to pursue his PhD with Professor Jay H. Lee and moved to Georgia Institute of Technology in August 2000. His PhD thesis was titled "modeling, analysis and control of nonlinear switching systems." He will join University of Delaware in January 2005 to pursue his postdoctoral research in multi-scale modeling with Professor Dionisios Vlachos. His research interests include Model Predictive Control, multi-scale modeling of reactive flows, microkinetic modeling, and nonlinear analysis and control of fuel processing systems.