ABSTRACT

BEELER, SCOTT COLVIN. Modeling and Control of Thin Film Growth in a Chemical Vapor Deposition Reactor. (Under the direction of Hien T. Tran.)

This work describes the development of a mathematical model of a high-pressure chemical vapor deposition (HPCVD) reactor and nonlinear feedback methodologies for control of the growth of thin films in this reactor. Precise control of the film thickness and composition is highly desirable, making real-time control of the deposition process very important. The source vapor species transport is modeled by the standard gas dynamics partial differential equations, with species decomposition reactions, reduced down to a small number of ordinary differential equations through use of the proper orthogonal decomposition technique. This system is coupled with a reduced order model of the reactions on the surface involved in the source vapor decomposition and film deposition on the substrate wafer. Also modeled is the real-time observation technique used to obtain a partial measurement of the deposition process.

The utilization of reduced order models greatly simplifies the mathematical formulation of the physical process so it can be solved quickly enough to be used for real-time model-based feedback control. This control problem is fairly complicated, however, because the surface reactions render the model nonlinear. Several control methodologies for nonlinear systems are studied in this work to determine which performs best on test examples similar to the HPCVD problem. One chosen method is extended to a tracking control to force certain film growth properties to follow desired trajectories. The nonlinear control method is used also in the development of a state estimator which uses the nonlinear partial observation of the nonlinear system to create an estimate of the actual state, which the feedback control formula then can use to guide the HPCVD system. The nonlinear tracking control and estimator techniques are implemented on the HPCVD model and the results analyzed as to the effectiveness of the reduced order model and nonlinear control.

MODELING AND CONTROL OF THIN FILM GROWTH IN A CHEMICAL VAPOR DEPOSITION REACTOR

BY

Scott Colvin Beeler

A DISSERTATION SUBMITTED TO THE GRADUATE FACULTY OF

NORTH CAROLINA STATE UNIVERSITY

IN PARTIAL FULFILLMENT OF THE

REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

APPLIED MATHEMATICS

RALEIGH

October 2000

APPROVED BY:

H. T. Tran	H. T. Banks
CHAIR OF ADVISORY COMMITTEE	
P. A. Gremaud	K. J. Bachmann

Biography

Scott Colvin Beeler was born and raised in Palo Alto, California, and attended Gunn High School there. He graduated magna cum laude from Pomona College, California, in 1995 with a Bachelor of Arts degree, double majoring in Mathematics and Music. He received his Doctorate of Philosophy in Applied Mathematics from North Carolina State University in December 2000, working under the direction of Dr. Hien Tran.

Acknowledgements

I would like to thank first and foremost my advisor, Dr. Hien Tran, for the uncountable hours of guidance he has given me, always available to help when I would drop in with no warning. He has been an excellent mentor, teacher, and friend with whatever support and assistance I needed. I also thank Dr. H. T. Banks, for his guidance when I was first starting my study at N.C. State, and for the regular pushes he has given me ever since then. I greatly appreciate the other members of my advisory committee, Dr. Pierre Gremaud and Dr. Kazufumi Ito, for their interest, comments and advice on my research.

Others besides my committee have been indispensable to me in my research work. I greatly appreciate all of my collaborators, especially Dr. Grace Kepler for her cooperation and support in many aspects of this research project, and Dr. Nikolaus Dietz and Dr. Klaus Bachmann in the Departments of Materials Science and Physics for all the help I ever needed with the experimental side of the project.

In addition to these I would like to thank the Department of Mathematics as a whole for the wonderful supportive environment it is, from the professors who I have taken classes from or just talked with in more casual circumstances, to the staff who always have the answers to the questions I always have, to my fellow graduate students who have always been good friends and colleagues. I am grateful also to the professors who guided my study of mathematics at Pomona College, especially my advisors there, Dr. Adolfo Rumbos and Dr. Richard Elderkin.

Financial support for this research project has been given by the Department of Defense through DOD-MURI Grant No. F49620-95-1-0447, and financial support for my personal research study at N.C. State has been given by a National Science Foundation (NSF) Graduate Research Traineeship.

Finally, I would like to thank my family for all the support they have given me; my brothers for their ever-present friendship, and my parents for giving me freedom when I wanted it, guidance when I needed it, and their love and encouragement always.

Table of Contents

List of Tables

List of Figures

Chapter 1

Background and Overview

Chemical vapor deposition (CVD) is a technique used to grow very thin films with certain desired properties, involving the deposition of source vapors onto a heated substrate surface where they then react chemically to form the desired material. This process is used in the manufacture of many computer hardware products, including high-speed (GaAs) integrated circuits, transistors, and DRAM chips, as well as UV detectors and green and blue light emitting diodes. A less well-known application is in high-performance electrostatic loudspeakers. Precise control of the film layer thickness and composition is extremely important, and the increasing demands on the precision of the desired properties make real-time feedback control of the CVD process very desirable [1, 2, 3, 4]. My research work at North Carolina State University has been in collaboration with members of the Materials Science, Physics, and Chemical Engineering Departments, as well as the Mathematics Department, in developing a high-pressure CVD reactor and the real-time sensing and control techniques to use on the film growth in this reactor.

Low-pressure chemical vapor deposition processes are the preferred choice for manufacturing many of the devices mentioned above. Previous work within this research group has successfully implemented feedback control of film thickness and composition in GaP/$Ga_{1-x}In_xP$ films, during experiments in a low-pressure pulsed chemical beam epitaxy (PCBE) reactor using real-time optical monitoring by p-polarized reflectance spectroscopy (PRS) sensing [4].

However, there are some materials (such as InN or $Ga_{1-x}In_xN$ films) which have potential industrial uses, but cannot be effectively produced at desirable temperatures under low-pressure conditions. Extending the CVD procedure to higher pressures increases our ability to control the thermal decomposition of certain source gases, and expands the range of compositions which can be produced at optimal process temperatures. This has applications to flat panel displays covering

the entire visible wavelength range, and optoelectronics in the visible to UV wavelength range, as well as radiation-resistant high power electronics. In addition, higher pressures give the advantage of a fuller ability to intentionally introduce controlled defects into the film or dope the film with impurities (for example, to give the film a positive charge, in the case of the speaker application). Control of defect chemistry/residual absorption and laser damage of nonlinear optical materials (such as $ZnGeP_2$) is also important for wave-guided nonlinear optical sensors and advanced optical parametric oscillators. Higher pressures can also result in faster film deposition and throughput, an advantage in time-intensive applications in the semiconductor industry. The difficulty in high-pressure chemical vapor deposition (HPCVD) is that it is significantly more difficult to control than the low-pressure process, as the higher pressure introduces source vapor gas flow dynamics in the place of low-pressure ballistic source vapor pulses.

The main focus of the CVD research project at N.C. State is the design and construction of a HPCVD reactor with real-time sensors to use in feedback control of the film growth process. A photograph of this Compact Hard Shell reactor is given in Figure 1.1; it consists of an outer cylindrical shell around a smaller cylindrical core, which contains a narrow rectangular-box-shaped flow channel. A photograph of the flow channel in one of the half-cylinder core parts is shown in Figure 1.2. The objects seen sticking out of the sides of the cylinder in Figure 1.1 are the measurement ports for observing the deposition process. The structure of this reactor is described in more detail in Section 3.2, with a discussion of the measurement techniques in Section 6.3. In collaboration with the rest of the group, we in the Mathematics Department have worked to effectively mathematically model the more complicated high-pressure deposition process. Another aspect of the project is the development of closed-loop control methods to use on the nonlinear system, including estimation of the state from the sensor measurements and tracking of desired properties such as film thickness and composition.

One part of the CVD model is the description of the surface kinetics, that is, the decomposition of source vapors deposited on the surface and their reactions forming the compound which is integrated into the growing film. These reactions are represented by a reduced order surface kinetics (ROSK) model which assumes that among the many reactions involved there are a small number of significant limiting steps. We developed the ROSK model and found values for the unknown parameters contained in it through comparison of the model with experimental observations of the surface deposition process at low pressure by p-polarized reflectance spectroscopy (PRS). This model, how it fits the experimental data, and the parameter identification process are described in Chapter 2.

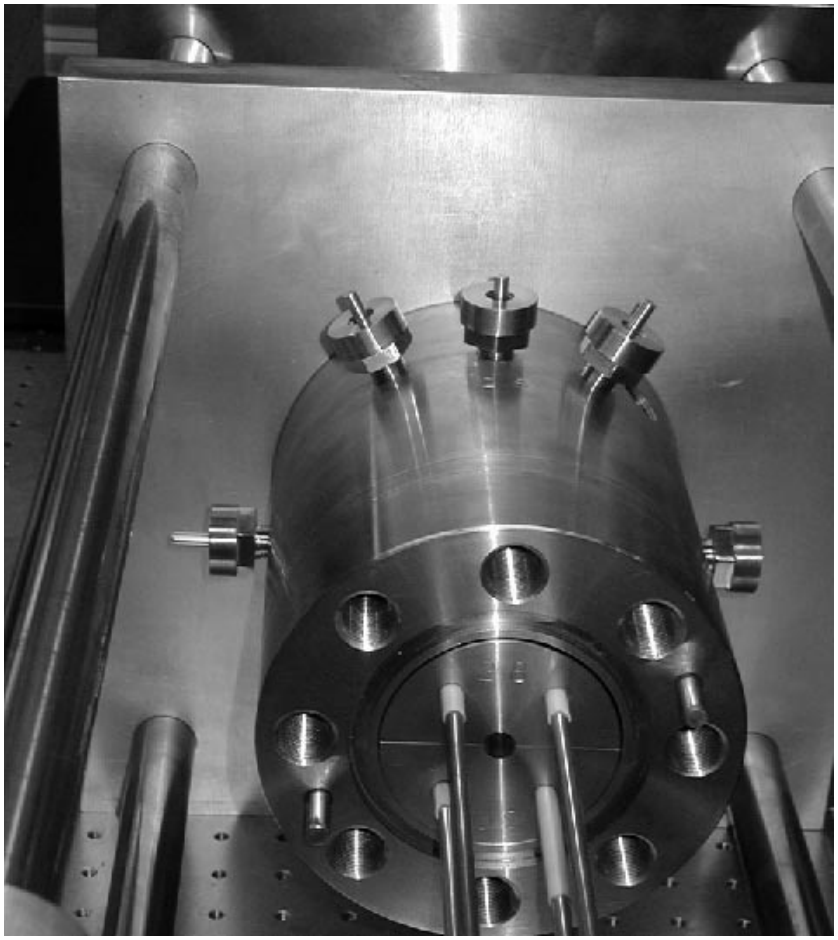

Figure 1.1: Photograph of the exterior of the Compact Hard Shell reactor.

The second part of the HPCVD process is the gas dynamics present in the high-pressure reactor. At low pressure the source vapor pulses are assumed to be ballistic beams which travel directly to the surface, but at high pressure this does not hold. The source vapors travel from the entrance of the reactor to the surface in a carrier gas at pressures of up to 100 atmospheres and so the mathematical model describing this process must include the system of partial differential equations representing the flow dynamics. A dilute approximation is used in our work, leading to a quasi-linear model with steady-state nonlinear continuity, momentum and energy equations being separated from the transient linear species equations. When solved numerically by a finite element, finite difference, or spectral method, the system of linear partial differential equations for the species mass fractions will

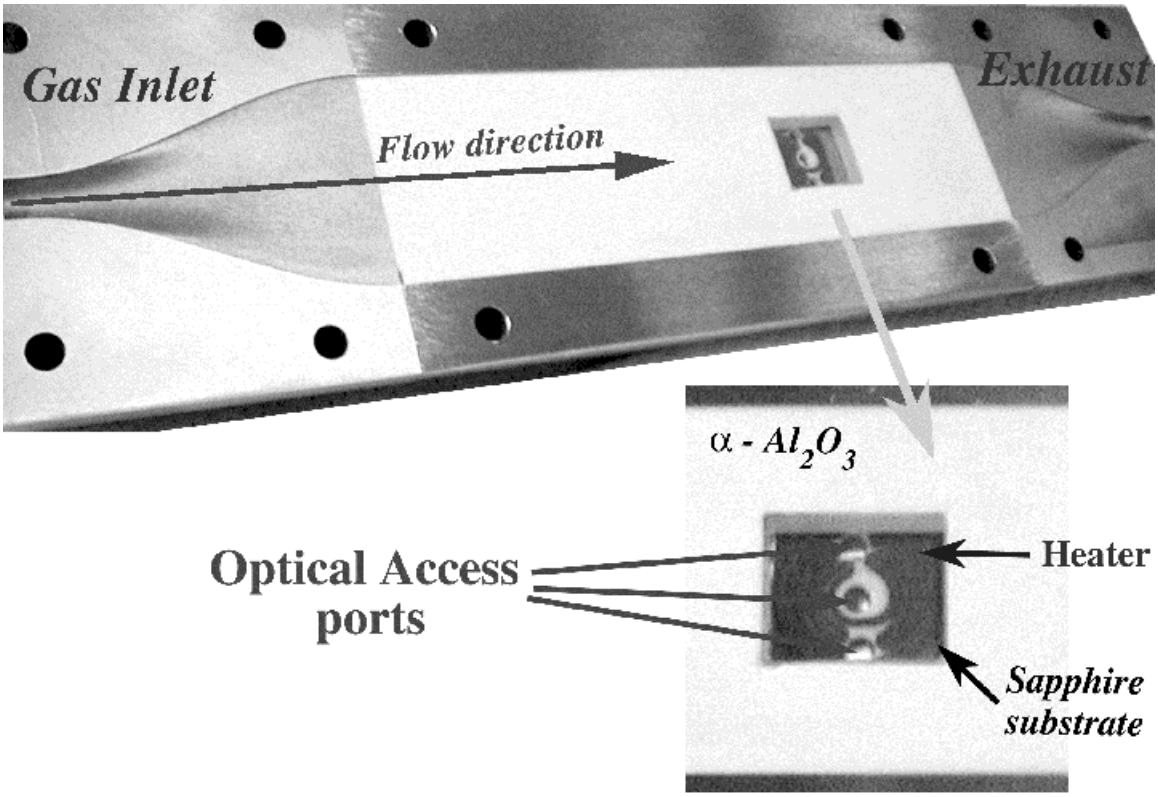

Figure 1.2: Photograph of the gas flow channel in the Compact Hard Shell reactor.

result in a very large system of ordinary differential equations. This makes real-time model-based feedback control impossible, so instead we obtain a set of basis functions specific to this problem, by using the reduced order method known as proper orthogonal decomposition (POD). In earlier work, members of the N.C. State research group have shown that the POD basis can be used to represent the species flow dynamics in another HPCVD reactor very efficiently and to compute an optimal open-loop control [5]. It has also been used to implement feedback control of HPCVD gas phase species transport to track a desired flux to the surface [6, 7]. Chapter 3 will describe the gas phase model, including species reactions and boundary conditions, and the use of the POD method for finding the reduced basis.

The full HPCVD model we will use here includes both the gas phase species transport and the surface dynamics, with the surface flux from the gas phase linking the two by becoming the source flux in the surface model. With the two parts combined, the model is nonlinear due to reactions

on the surface, so a nonlinear feedback control method must be used. Unlike for linear systems, there is no standard method for feedback control of nonlinear systems. Therefore in Chapter 4 we compare the performance of several methods from the literature on some simpler test problems, to gain information with which to choose a particular method to use on the HPCVD problem.

The HPCVD problem also involves more than simple feedback control, however. We want to track a given film thickness growth curve, so we will need a tracking control instead of a stabilizing control which simply forces the system to zero. There are also only nonlinear partial measurements of the growth process available, as will be described in Section 6.3. Lacking full state knowledge, a state estimator is needed to reconstruct an estimated value of the full state from these measurements. In Chapter 5 we take one of the nonlinear feedback controls studied in Chapter 4, the state-dependent Riccati equation (SDRE) method, and extend it into a method for feedback tracking control, as well as use it in a nonlinear state estimator. These new methods are tested on some simple examples and their results are compared with those of established linear control methods.

Chapter 6 describes the application of the nonlinear tracking control and state estimation methods to the combined gas-phase and surface HPCVD model. The goal will be to have the growing film thickness track a chosen desired profile. This chapter details the linking of the two parts of the model as well as the formulation of the control problem, including the measurement technique and the desired tracking signal. Results of the control implementation are given and analyzed, to study the effectiveness of the POD representation of the species flow, the nonlinear control using it and the ROSK model, and the state estimation process using an optical absorption measurement for partial observation of the state. Figure 1.3 shows the structure of this film growth control problem, the various components of which will be discussed in the following chapters. A summary of all the research work contained here is given in Chapter 7, with an evaluation of the techniques developed and recommendations for future research directions.

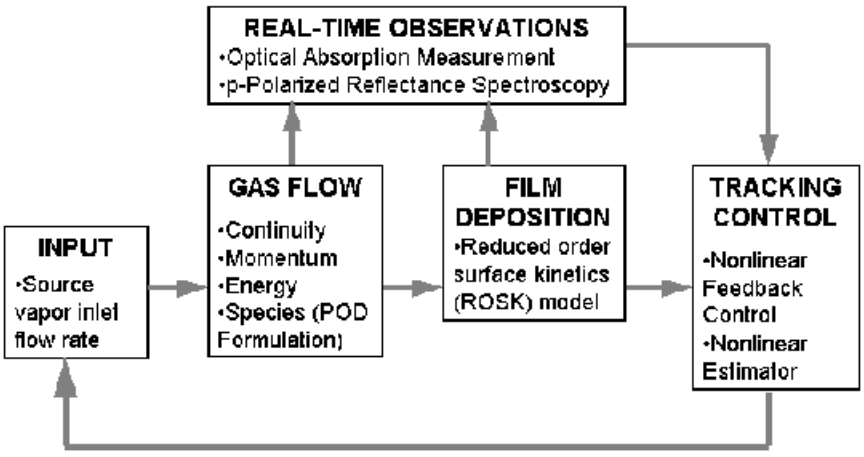

Figure 1.3: Outline of the structure of the film growth control problem.

Chapter 2

Reduced Order Modeling of the Surface Kinetics of Thin Film Growth

2.1 Introduction

Understanding and controlling thin film growth is a difficult task, because little is known about the chemical reaction pathways and reaction kinetics parameters involved in the decomposition of metalorganic (MO) precursors as source vapors in chemical vapor deposition and their incorporation into the film. In this chapter these III-V compound/silicon heterostructure growth processes are described by a reduced order surface kinetics (ROSK) model [8], which reduces the many surface reactions to a small number of important steps. We will use a series of experiments monitoring gallium phosphide (GaP) film growth to find estimated values for the unknown constants in the ROSK model.

To monitor and control the deposition process with stringent tolerances with respect to controlled thickness and composition of ultrathin layers requires the development of techniques that follow the deposition process with submonolayer resolution. These demands led to the development of surface-sensitive real-time optical sensors [9] that are able to move the monitoring and control point close to the point where the growth occurs, which in a chemical beam epitaxy process is the surface reaction layer, which is built up by physisorbed and chemisorbed precursor fragments between the ambient and the film interface. A main challenge in applying optical probe techniques to real-time characterization of thin film growth is in relating the surface chemistry processes that drive the growth process to film growth properties, such as composition, instantaneous growth rate or structural layer quality. As illustrated in Figure 2.1, in deposition four primary regions are involved:

(1) the ambient, (2) the surface reaction layer, which consists of species physisorbed or chemisorbed to the surface in dynamic equilibrium with both ambient and surface, (3) the surface itself, and (4) the near-surface region that can be defined as consisting of the outermost several atomic layers of the fabricated sample. Presently most characterization techniques are aimed towards accurately

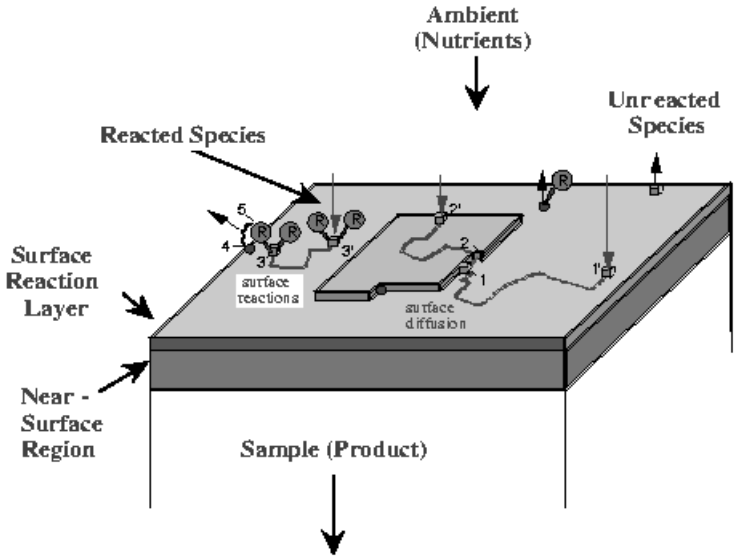

Figure 2.1: The four primary regions involved in thin film deposition.

measuring ambient process parameters, such as pressure, flux or temperature. This limits their ability to deal with complex nonlinear surface chemistry processes, where the surface plays an integral role in the precursor decomposition pathways and small changes in the ambient composition can affect the growth substantially. To this end, we have developed and explored p-polarized reflectance spectroscopy (PRS) [10, 11, 12] as a highly surface-sensitive monitoring technique, which allows us to follow the surface reaction kinetics closely under steady-state growth conditions.

To study the ROSK model and try to match its behavior to the actual growth process, we use PRS-monitored film growth experiments under various conditions. These are pulsed chemical beam epitaxy (PCBE) experiments done at low pressure, growing GaP heterostructures on Si(001) substrates. The low-pressure nature of the experiments allows us to focus only on the surface kinetics without the gas dynamics of the source vapor transport in the high-pressure CVD reactor becoming a factor. The high surface sensitivity of PRS allows us to follow alterations in the composition and thickness of the surface reaction layer (SRL) as they appear during the periodic pulsed precursor

supply. The linkage of the PRS response to the reduced order surface kinetics model provides the basis for the estimation of parameters in the ROSK model, and gives insights into the organometallic precursor decomposition and growth kinetics, allowing us to adjust the model to more accurately represent the deposition process. The PRS measurement technique has been applied to closed-loop control of deposition processes at low pressure (PCBE) [8]. A variation on PRS will also be used as one of the measurements in the physical high-pressure CVD reactor (though not in the model described here), for real-time observation and control of the film growth.

We will give a brief background on the experimental growth and monitoring conditions in the PCBE reactor in Section 2.2 and show results obtained by PRS during real-time monitoring of heteroepitaxial growth of GaP on Si substrates. In Section 2.3 we introduce the model used to simulate the surface kinetics and the PRS measurements of them. We describe there the link of the PR response to the simulation parameters within the ROSK model for describing the decomposition kinetics of the involved organometallic precursors. The process of identifying these parameters is explained in Section 2.4, and in Section 2.5 the results of the parameter identification through various experiments are analyzed. In this way we establish and validate surface reaction kinetics parameters, and advance our understanding of fundamental chemistry processes in thin film growth using organometallic precursors. We give some concluding remarks on the modeling of CVD surface kinetics in Section 2.6.

2.2 Experimental Setup and PRS Measurement Results

For monitoring both the bulk and surface properties during heteroepitaxial growth of GaP on Si, *p*-polarized reflectance spectroscopy (PRS) has been integrated into a pulsed chemical beam epitaxy (PCBE) system schematically shown in Figure 2.2. In PCBE, the surface of the substrate is exposed to pulsed ballistic beams of $(C_4H_9)PH_2$ (tertiary-butyl phosphine, or TBP) and $Ga(C_2H_5)_3$ (triethylgallium, or TEG) at typically $350 - 450^\circ$ C to accomplish nucleation and overgrowth of the silicon by an epitaxial GaP film. For PRS and laser light scattering (LLS) measurements we employ *p*-polarized light beams at two angles of incidence (PR70: $\phi = 71.5^\circ$ and PR75: $\phi = 75.2^\circ$) using the wavelength $\lambda = 632.8$ nm and Glan-Thompson prisms, as illustrated in Figure 2.3. Further details on the experimental conditions are given in previous publications [8, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22].

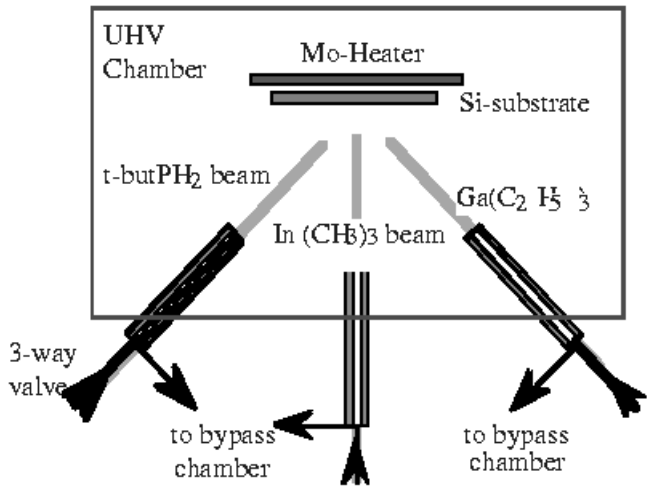

Figure 2.2: Setup of the PCBE system for III-V compound growth.

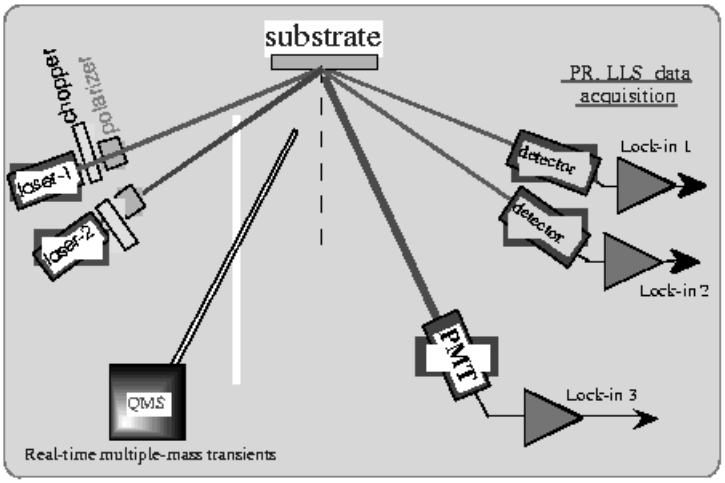

Figure 2.3: Setup of growth monitoring by PRS, LLS, and quadrupole mass spectroscopy (QMS).

During the preconditioning period, the PR signals change according to the temperature dependency of the substrate dielectric function. The PR signals are used to verify independent temperature measurements and to calibrate the actual surface temperature. A constant flow of Palladium purified H_2 (10 sccm) is introduced into the growth chamber during the preconditioning as well as during the growth period. The background pressure in the growth system is $< 10^{-9}$ Torr, increasing to 5×10^{-5} Torr during pregrowth and to 2×10^{-4} Torr during growth.

Figure 2.4 shows the PR and LLS signals during heteroepitaxial growth of GaP on Si(001). A 1200 s preconditioning period begins the observed measurements. After initiating growth at

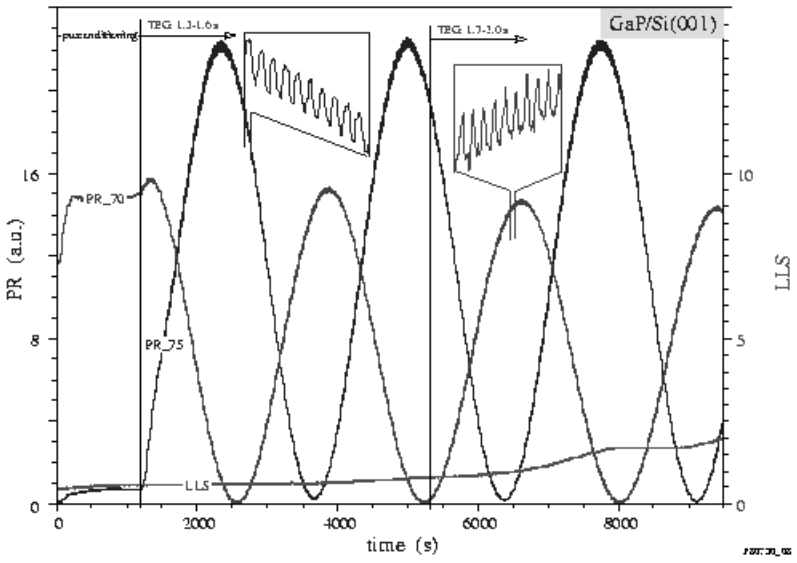

Figure 2.4: PRS and LLS monitoring of heteroepitaxial GaP growth under PCBE conditions.

1200 s, minima and maxima are observed in the time evolution of the PR signals due to interference phenomena as the film thickness increases. It should be noted that the maxima and minima of the two signals are inverted, which is due to the fact that one angle of incidence (PR75) is above – and the other (PR70) below – the pseudo-Brewster angle of the growing film. Superimposed on the interference oscillations of the reflectance is a fine structure that is strongly correlated to the time sequence of the supply of precursors employed during the steady-state growth conditions. The two insets in Figure 2.4 show enlargements of the fine structure evolutions for 30 s of growth for PR75 and PR70, respectively.

To analyze the surface reaction kinetics and validate simulations performed using the ROSK

model to be presented in Section 2.3, we varied two experimental parameters: (i) the position of the TEG pulse of 300 ms length within the precursor cycle sequence and (ii) the TEG flow rate during the pulse. Each growth condition was carried out and monitored for at least one and a half interference oscillations in order to get stable steady-state growth and to gather sufficient information to analyze and compare with simulations of the growth process.

The correlation of the fine structure evolution with the pulsing sequence of the precursor supply is shown in more detail in Figure 2.5. In this figure the PR75 response is taken during steady-state

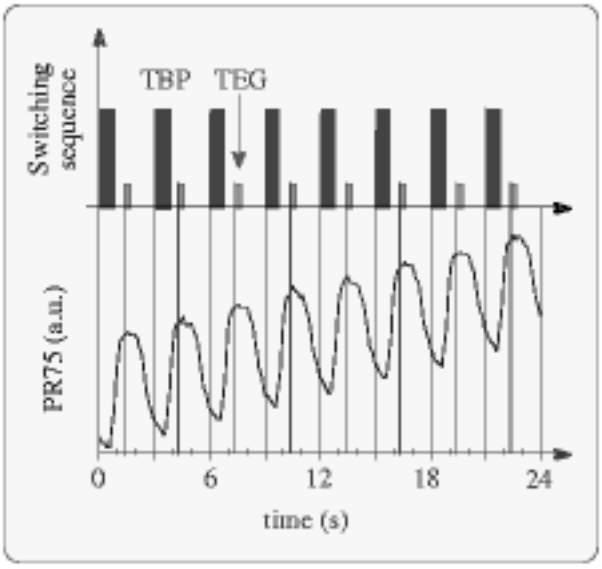

Figure 2.5: PR75 response to periodic TBP and TEG precursor pulses.

growth on the rising flank of an interference oscillation, using a pulse cycle sequence of 3 s, a TBP pulse from 0.0 to 0.8 s, a TEG pulse from 1.3 to 1.6 s and continuous hydrogen flow during the whole sequence. In the first set of experiments, the flow rates and pulse durations of TBP (800 ms at 0.907 sccm) and TEG (300 ms at 0.04 sccm) and the start position of the TBP pulse (at 0.0 s) were kept constant, and only the start position of the TEG pulse was varied, from 0.9 s up to 2.3 s (increased in steps of 0.2 s). The effect on the fine structure evolution is shown in Figure 2.6, where the starting point of the TEG pulse is marked by an arrow. This influence of TEG pulse position on the PR response will be explained more fully in Section 2.5. For comparison, all PR responses are taken at the same intensity/reflectance level on a rising flank of an interference oscillation. We note that the exposure times as well as the precursor fluxes are identical for each trace shown in

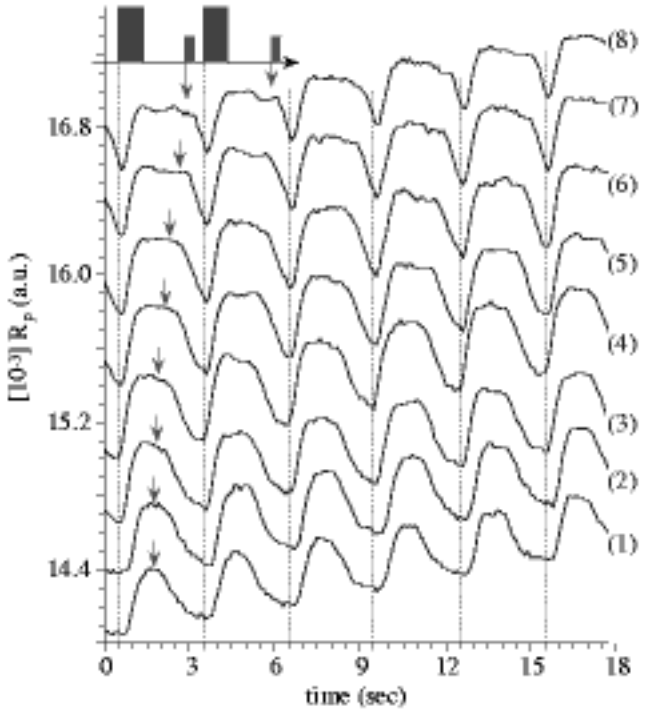

Figure 2.6: PR75 responses for various TEG positions within the cycle sequence.

Figure 2.6.

In the second set of experiments, the changes in surface reaction kinetics and growth are evaluated for TBP:TEG flow ratios between 18 and 30. Figure 2.7 shows the PR and LLS signals during heteroepitaxial growth of GaP on Si(001) for three TEG flow settings of 0.05, 0.04 and 0.03 sccm (in a pulse at 1.3-1.6 s), with a TBP pulse of 0.907 sccm at 0.0-0.8 s, in a pulse cycle sequence 3 s in duration. With decreasing TEG flow, the spacing of the interference oscillations widens according to the reduced growth rate. More details including comparisons with the results of simulations are given in Section 2.5.

2.3 Modeling of Surface Reactions and PRS Measurements

We represent the structure of a growing heteroepitaxial film during chemical vapor deposition with a four-layer medium model consisting of: (1) ambient, (2) surface reaction layer (SRL), (3) film and (4) substrate. We consider here GaP film growth on a Si substrate. The complex reflectivity

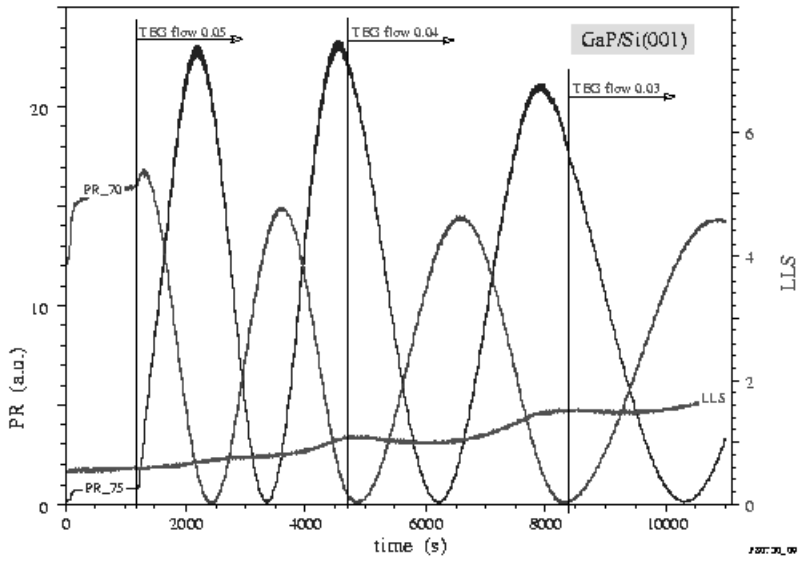

Figure 2.7: PRS and LLS responses to various TEG flow rates (changing at marked positions).

coefficient for p-polarized incident light, given a four-layer stack, is [23]

$$r_p = \frac{r_{12}(1 + r_{23}r_{34}e^{-2i\beta_3}) + (r_{23} + r_{34}e^{-2i\beta_3})e^{-2i\beta_2}}{(1 + r_{23}r_{34}e^{-2i\beta_3}) + r_{12}(r_{23} + r_{34}e^{-2i\beta_3})e^{-2i\beta_2}}, \tag{2.1}$$

where the Fresnel coefficients $r_{k(k+1)}$ ($k = 1$, 2, and 3) for the interfaces 1-2, 2-3, and 3-4 are given by [24]

$$r_{k(k+1)} = \frac{\epsilon_{k+1}\sqrt{\epsilon_k - \epsilon_1 \sin^2\phi_1} - \epsilon_k\sqrt{\epsilon_{k+1} - \epsilon_1 \sin^2\phi_1}}{\epsilon_{k+1}\sqrt{\epsilon_k - \epsilon_1 \sin^2\phi_1} + \epsilon_k\sqrt{\epsilon_{k+1} - \epsilon_1 \sin^2\phi_1}}, \tag{2.2}$$

and the phase shifts β_k for the SRL ($k = 2$) and the growing film ($k = 3$) are given by

$$\beta_k = \frac{2\pi}{\lambda} d_k \sqrt{\epsilon_k - \epsilon_1 \sin^2\phi_1}. \tag{2.3}$$

Using equations (2.1)-(2.3), the reflectivity coefficient r_p is a function of d_2 and d_3 (the thicknesses of the SRL and film respectively), ϵ_1, ϵ_2, ϵ_3 and ϵ_4 (the complex dielectric functions of the ambient, SRL, film and substrate, respectively), and ϕ_1 and λ. Here ϕ_1 denotes the angle of incidence and λ the wavelength of the impinging laser light [23].

The values of ϵ_1, ϵ_3, ϵ_4, ϕ_1 and λ are constant in time, but ϵ_2, d_2 and d_3 vary in time as the film grows and the SRL composition and thickness change. To understand how these values change, we need a representative model of the chemical kinetics of the SRL, which approximates the pyrolysis of the primary source vapors and has been discussed in detail elsewhere [25, 26, 27]. For TBP and

TEG as source vapors forming GaP, we employ a reduced order surface kinetics (ROSK) model [8]. The ROSK model makes the simplifying assumption that the many reactions which make up the TBP pyrolysis are combined into one step, the reactions which make up the TEG decomposition are combined into two steps, and the formation of GaP is one final step. The process is driven by a periodic source vapor cycle as described in Section 2.2.

Thus the kinetic model representing the reactions in the SRL is given by the following system of ordinary differential equations:

$$\frac{d}{dt}n_1(t) = S_1(t) - k_1 n_1(t) - k_{GaP} n_1(t) n_3(t)/10^{-8}\text{mol} \tag{2.4}$$

$$\frac{d}{dt}n_2(t) = S_2(t) - k_2 n_2(t) - k_3 n_2(t) \tag{2.5}$$

$$\frac{d}{dt}n_3(t) = k_3 n_2(t) - k_4 n_3(t) - k_{GaP} n_1(t) n_3(t)/10^{-8}\text{mol} \tag{2.6}$$

$$\frac{d}{dt}n_4(t) = k_{GaP} n_1(t) n_3(t)/10^{-8}\text{mol}. \tag{2.7}$$

The variables n_1, n_2 and n_3 represent the number of moles of the components of the SRL, with n_1 being active surface phosphorus fragments, the single stage in the TBP decomposition. An intermediate stage in the TEG decomposition is represented by n_2, and we will consider this to be diethylgallium (DEG), while n_3 represents the final stage which we will consider to be monoethylgallium (MEG) and active gallium fragments. In equation (2.4) the change in active phosphorus fragments is written as the sum of a source term S_1, a desorption loss term $-k_1 n_1$, and a nonlinear reaction term forming GaP. The second differential equation, (2.5), which describes the change in DEG, contains a source term S_2, a desorption loss term $-k_2 n_2$, and a term of decomposition into MEG and active gallium fragments. Equation (2.6) (change in MEG and active surface gallium fragments) has a term of creation, a desorption loss term, and a reaction term forming GaP. The variable n_4, in equation (2.7), represents the number of moles of created GaP integrated into the deposited film layer. This equation contains only the single nonlinear reaction term for the formation of GaP from active surface Ga and P and has to account for any surface activation processes.

The source terms in the differential equations are based on the source vapor pulses. More specifically, we model the first source term by the following expression:

$$S_1(t) = \frac{P_1(t)\gamma\beta_{TBP}}{V_{TBP}}, \tag{2.8}$$

where $P_1(t)$ is the source vapor flow rate. V_{TBP} is the molar volume of TBP and the constant β_{TBP} is the sticking coefficient of TBP. The geometrical parameter γ represents how much of the source

vapors actually hit the surface of the Si substrate wafer (a constant dependent on the structure of the reactor). Similarly, the second source term is represented by

$$S_2(t) = \frac{P_2(t)\gamma\beta_{TEG}}{V_{TEG}}, \tag{2.9}$$

with corresponding $P_2(t)$, V_{TEG} and β_{TEG} for the TEG pulse, and the same constant γ. For each source term we are using pulsed flow, with a constant flow rate between start and stop times (and zero flow elsewhere), as described in Section 2.2. There is a small time difference between the switching on (or off) of the pulse and the start (or stop) of the source vapors at the surface. This is caused by the time needed for the source vapor gates to open or close and for the vapors to travel to the surface. We account for it with a parameter *delay*, so that for a source vapor pulse set to start at t_{on} and stop at t_{off}, the source vapors will actually reach the surface starting at $t_{\text{on}} + delay$ and stopping at $t_{\text{off}} + delay$. The delay was estimated to be 0.72 s using the parameter identification process to be described in Section 2.4.

The system of differential equations (2.4)-(2.7), together with the source terms (2.8) and (2.9) and appropriate initial conditions, can be solved numerically for the number of moles n_1, n_2, n_3 and n_4 at all times during the growth process. From these solutions, the film and SRL thicknesses are found by the equations

$$d_3(t) = \frac{V_{GaP}}{A} n_4(t), \tag{2.10}$$

$$d_2(t) = \frac{\alpha_{SRL}}{A} \left[V_1 n_1(t) + V_2 n_2(t) + V_3 n_3(t) \right], \tag{2.11}$$

and the effective dielectric function of the SRL is given by

$$\epsilon_2(t) = 1 + \left[\frac{n_1(t)}{\sum_{k=1}^{3} n_k(t)} F_1 + \frac{n_2(t)}{\sum_{k=1}^{3} n_k(t)} F_2 + \frac{n_3(t)}{\sum_{k=1}^{3} n_k(t)} F_3 \right], \tag{2.12}$$

which is derived from the Sellmeier equation [28]. In the above three equations, A is the surface area of the Si wafer, the values V_k are the molar volumes of the components n_k, and V_{GaP} is the molar volume of GaP. The parameters F_k are the complex optical responses of the components of the SRL and α_{SRL} is an effective SRL thickness parameter representing the fraction of the SRL that contributes to the reflectance behavior (as opposed to that which is floating loose on top of the SRL and does not affect the light beams). With the values of the time-dependent parameters ϵ_2, d_2 and d_3 found by equations (2.10)- (2.12), and with the constant parameters ϵ_1, ϵ_2, ϵ_4, ϕ_1 and λ, the reflectivity coefficient r_p can be computed from equations (2.1)-(2.3). From r_p, we then find the value that is actually measured in the experiments by computing the reflectance $R_p = |r_p|^2$.

In the equations described in this section, ϕ_1, λ, ϵ_1, V_1, V_2, V_3, V_{GaP}, V_{TBP}, V_{TEG}, A, β_1, β_2, α_{SRL}, and all start/stop times and flow rates contributing to P_1 and P_2 are known quantities. The values of the dielectric functions ϵ_3 and ϵ_4, the rate constants k_1, k_2, k_3, k_4 and k_{GaP}, the optical responses F_1, F_2 and F_3, the geometrical parameter γ, and *delay* are not known. Our work in Sections 2.4 and 2.5 is to find values of these parameters so that the mathematical model most closely matches experimental results.

2.4 Parameter Identification Problem

Here we formulate the inverse least squares problem used to identify the unknown parameters in the ROSK model. We want the set of parameters which results in the simulated reflectance (from the mathematical model described last section) most closely fitting the experimental data. Specifically, we are looking for the vector of parameters $\overrightarrow{q} = [F_1, F_2, F_3, k_1, k_2, k_3, k_4, k_{GaP}, \gamma, delay]$ that minimizes the cost function

$$J(\overrightarrow{q}) = \sqrt{\sum_i \left[R_{\text{exp}}(t_i) - R_{\text{calc}}(t_i, \overrightarrow{q})\right]^2}.$$

Here $R_{\text{exp}}(t_i)$ is the experimental PRS data at the measurement times t_i and $R_{\text{calc}}(t_i, \overrightarrow{q})$ is the simulated data calculated at the same times using the parameter set $\overrightarrow{q}$.

We do not include ϵ_3 and ϵ_4 in $\overrightarrow{q}$ because larger numbers of parameters make the minimization process increasingly difficult. We can remove these two parameters from the above parameter estimation problem by formulating a separate simpler estimation problem. In particular, we use a three-layer stack as a simplified model of the growing film, removing the SRL from consideration and leaving just the ambient, film and substrate layers. The formula for calculating the reflectance for a three-layer stack analytically is given by

$$r_{3,p} = \frac{r_{13} + r_{34}e^{-2i\beta_3}}{1 + r_{13}r_{34}e^{-2i\beta_3}}, \tag{2.13}$$

where r_{13} and r_{34} are the Fresnel coefficients for the reflection from interfaces 1-3 and 3-4 (now that layer 2 is removed), and β_3 is the phase shift for the film layer. These values are calculated by formulas analogous to (2.2) and (2.3).

To compare results from this formula with experimental results, we first remove the effects of the SRL from the experimental data by removing the small-amplitude fine structure oscillations modulated with the precursor cycle from the large-amplitude interference oscillations, which have a

periodicity of several hundreds of seconds. In order to remove the fine structure, first the curves on either side of the data forming an envelope around it must be found. The experimental version of the three-layer stack reflectance is then found by switching from one side of the envelope to the other where the fine structure "turns" from positive to negative (from adding to the three-layer stack reflectance to subtracting from it) or vice versa. This orientation of the fine structure is cyclical with the interference oscillations, either turning twice per oscillation or else not turning at all, in which case there is no switching between envelope sides. Figure 2.8 shows this extraction of the three-layer reflectance out of experimental data near a turning point. The three-layer reflectance

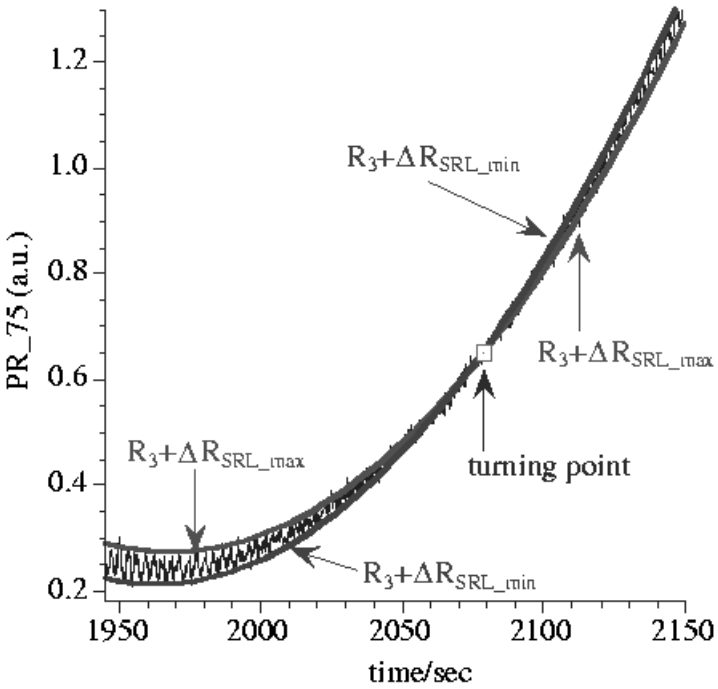

Figure 2.8: Extraction of reflectance envelope from PR75 data by removing fine structure.

plus the minimal influence from the SRL during a cycle is shown on one side of the data (switching sides at the turning point), while the other side represents the three-layer stack plus the maximal influence from the SRL during the cycle.

With this method of extracting the experimental three-layer stack reflectance $R_{3,\text{exp}}$, we can identify the parameters ϵ_3 and ϵ_4, as well as an average growth rate $\widehat{g}_r$ (used to find the film thickness at the times t_i), by comparing the calculated reflectance $R_{3,\text{calc}} = |r_{3,p}|^2$ from equation (2.13) to

$R_{3,\text{exp}}$. This is also done through an inverse least squares formulation by finding $\overrightarrow{r} = [\epsilon_3, \epsilon_4, \widehat{g}_r]$ that minimizes the cost function

$$J(\overrightarrow{r}) = \sqrt{\sum_i [R_{3,\text{exp}}(t_i) - R_{3,\text{calc}}(t_i, \overrightarrow{r})]^2}.$$

Once the values of ϵ_3 and ϵ_4 are found, they can be used in solving the four-layer stack parameter identification problem to find the unknown parameters $\overrightarrow{q} = [F_1, F_2, F_3, k_1, k_2, k_3, k_4, k_{GaP}, \gamma, delay]$.

2.5 Analysis of Results

Comparing measurements taken with the TEG pulse position varied while all other conditions are fixed, as described in Section 2.2 (see e.g., the fine structures shown in Figure 2.6), reveals several important characteristics in the fine structure. We will explain these features and show how the mathematical simulation of the growth process, using the reduced order surface kinetics model, replicates these features.

Looking at the fine structure (in the PR75 data), the most noticeable change with the TEG pulse position variation is the starting position of the downward slope (near the arrows marked in Figure 2.6) which is present in every data set but moves to later in the cycle as the TEG pulse moves to later in the cycle. This start of the downward slope, which is the only feature so dependent on the TEG pulse placement, clearly relates to the source TEG, the subsequent TEG defragmentation, and active gallium attachment on the surface.

In contrast, the starting position of the upward slope in the fine structure remains in the same place shortly after the start of the cycle, independently of the TEG pulse position. It can be related to the source TBP exposure, its defragmentation, and the formation of active phosphorus on the surface. Both starting positions are delayed by approximately 0.72 s after the start of the pulses. This delay is due to the time needed to open the source vapor gates and the time for the vapors to travel to the surface, as noted in the description of the model in Section 2.3.

The same upward and downward slopes and delay characteristics can be seen in Figure 2.9, where the fine structure evolutions of the simulated data are compared at the same points as the experimental data in Figure 2.6, also with the TEG start positions marked by an arrow. The gap between the downward and upward slopes can be analyzed by the full width at half maximum (FWHM), defined by the width between the times on the downward and upward slopes with values halfway between the maximum and minimum reflectance during that cycle. Figure 2.10 illustrates

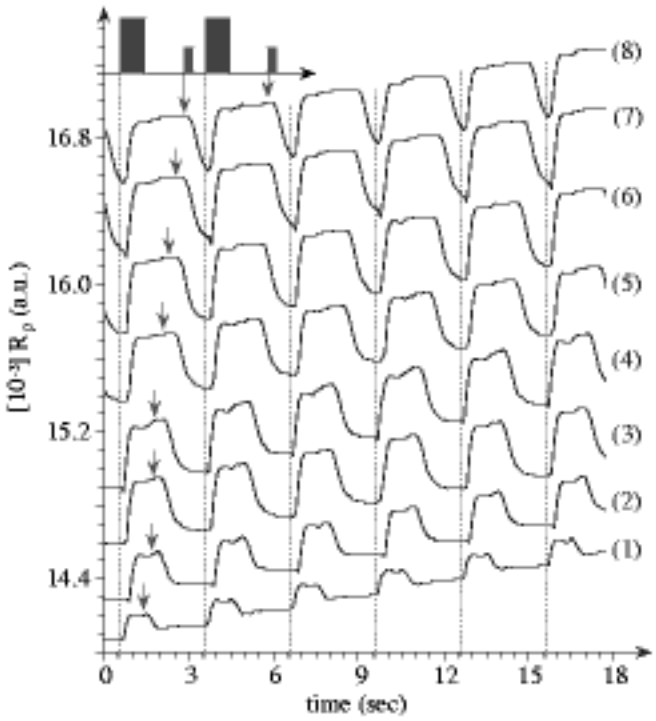

Figure 2.9: Simulated PR75 responses for various TEG positions within the cycle sequence.

how this width shrinks as the TEG pulse is moved towards the end of the cycle and closer to the next TBP pulse. This change, in both the experimental and calculated data, follows the pulse position nearly linearly.

The fine structure amplitude (the difference between maximum and minimum reflectance over a cycle) also changes slightly but clearly with the change in TEG pulse position. As shown in Figure 2.10, the amplitude is largest for TEG pulses near the middle of the range used. This can be explained as a result of the closeness of the TEG and TBP pulses. If the TEG is input soon after the TBP, there will be a large GaP formation reaction, leaving little or no active gallium to carry over to the next cycle. If the TEG comes in very late in the cycle, right before the next TBP pulse, there may not be time for the decomposition of all the TEG to gallium to occur before the GaP formation with the incoming phosphorus starts. With a more central TEG pulse, the phosphorus and gallium will each have the time to build up on the surface, in turn creating more extreme changes in the SRL thickness and composition and therefore a larger fine structure amplitude.

Note that this analysis of the fine structure is at a specific place on the interference oscillations,

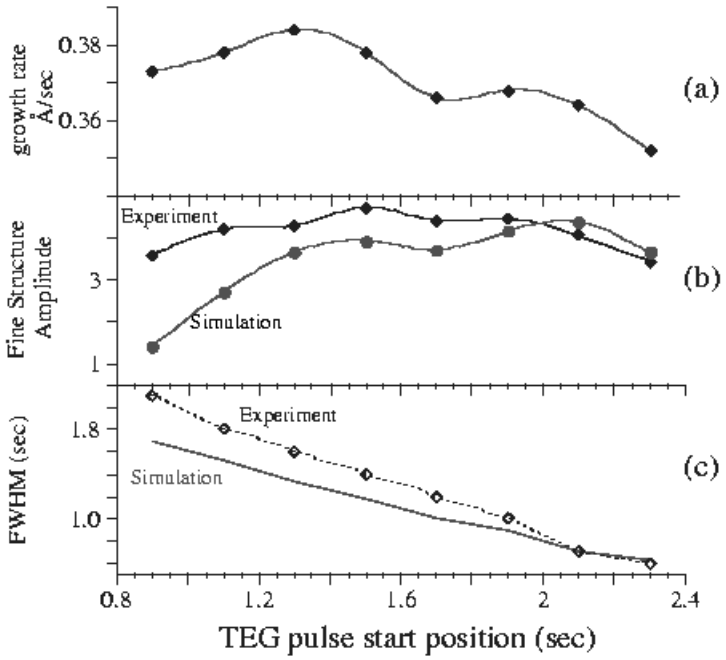

Figure 2.10: Properties of the PR75 responses as affected by the TEG start position.

fairly high on a rising flank. Other places, particularly on the other side of a turning point, will have different characteristics (for example, the TEG pulse may result in a jump upward and the TBP pulse in a jump downward).

One larger-scale feature of the reflectance data we can look at is the average film growth rate for the various TEG pulse positions, shown in Figure 2.10. The general downward slope can be explained in terms of the closeness of the two pulses. As the TEG pulse moves later in the cycle away from the TBP pulse there is less phosphorus to react with, so there is more active gallium left on the surface to be lost via desorption before the next cycle. The TEG pulse positions nearest the start of the cycle seem to be too close to the TBP pulse for the fastest growth rate, however. The incoming TEG and its defragmentation products may be partially blocked from the available active phosphorus in the SRL by TBP that failed to stick and/or desorbed phosphorus that is sitting loose on the surface.

Another large-scale characteristic feature of the data sets is the position of (or complete lack of) turning points in the fine structure. These come in pairs for every interference oscillation or not at all, as discussed in Section 2.4 (with a close-up of a turning point in Figure 2.8). The

turning point positions can be characterized by the derivative of the reflectance. The derivative amplitude is related to the periodic thickness changes in the SRL. This amplitude is minimized at the turning points, where the fine structure amplitude is smallest and so the reflectance curve is least steep. Figure 2.11 shows the close match between the experimental and calculated derivative amplitudes, which characterize the fine structure amplitudes as well as their turning point positions (using TEG pulse position 1.3-1.6 s). In earlier works [11, 12], we showed that the locations of these

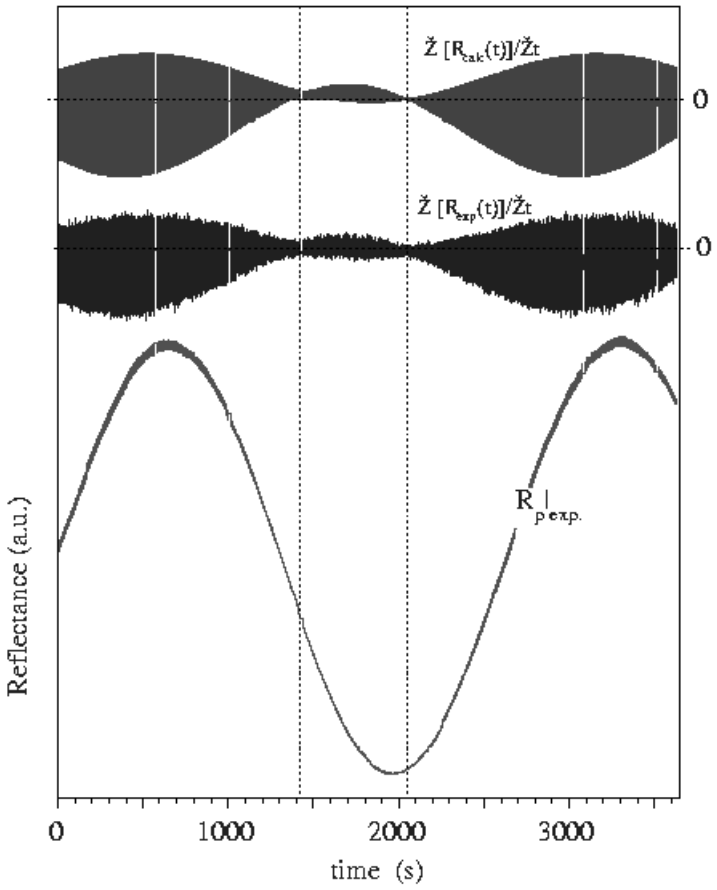

Figure 2.11: Simulated and experimental PR75 first derivatives and reflectances.

turning points change as a function of the SRL dielectric properties. The good agreement shown in Figure 2.11 indicates that the SRL dielectric properties were obtained correctly. The simulated PR75 reflectance matches the experimental reflectance shown in 2.11 so closely that with the two plotted on top of each other no differences can be seen.

The measurements taken with the TEG pulse position fixed but the flow rate varied (as described in Section 2.2) also correspond to what the model predicts. Examples of the fine structure (again for PR75) for the three TEG flow rates are shown in Figure 2.12 for both experimental data and simulated data. In contrast with the variation of the pulse position, here the shape of the fine

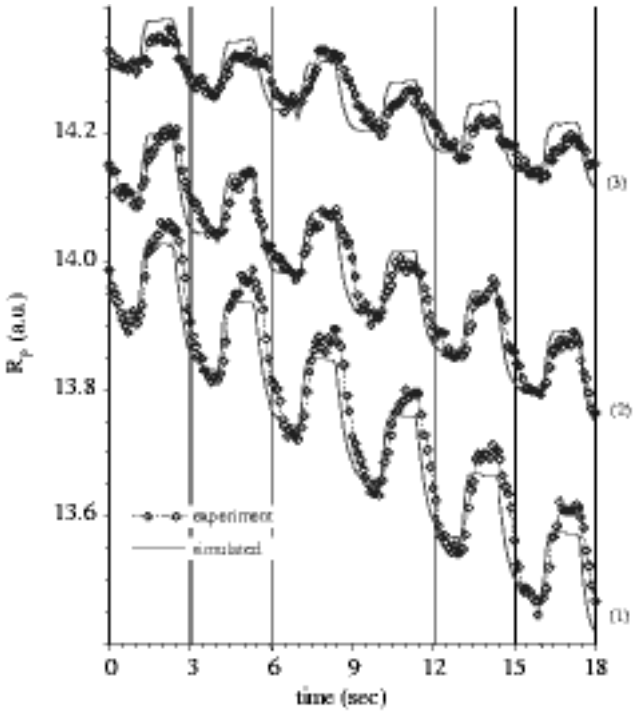

Figure 2.12: Simulated and experimental PR75 responses for various TEG flow rates.

structure remains the same, since the shape of the source vapor cycle is the same. The positions of changes in the slope remain constant due to the constant position of the TEG pulse. However, the amplitude of the fine structure does change, since as the TEG flow rate increases there will be more gallium deposited in the SRL, and this will have a larger effect on the reflectance. Increased TEG flow also results in a faster film growth rate, which causes steeper large-scale curves as seen in Figure 2.12 and faster interference oscillations as seen in Figure 2.7. Both the experimental and simulated data sets show these characteristics, and they agree with each other extremely well.

The steps in the generation of a set of simulated data which was used to compare with the experimental data presented in Figure 2.4 are shown in detail in Figure 2.13 for a TEG pulse of 1.3-1.6 s (and a TBP pulse of 0.0-0.8 s). The number of moles of the three SRL components are

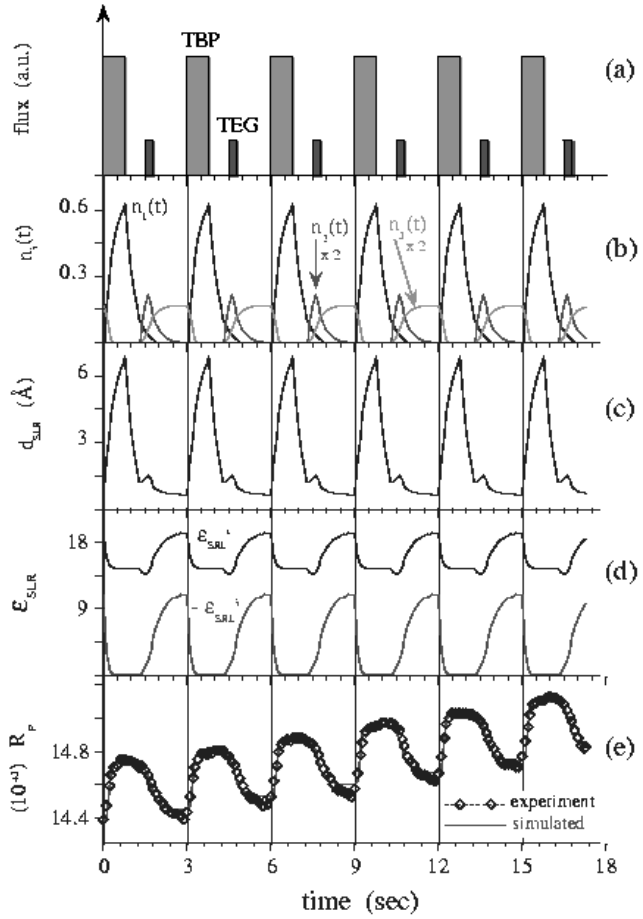

Figure 2.13: Construction in the model of simulated reflectance from a source pulse cycle.

the result of the source pulses and the ROSK model simulation. From the SRL components, the SRL thickness and dielectric function are found. These values then contribute to the calculated reflectance. Figure 2.13 shows how the arrival of gallium in the SRL causes the downward slope in the fine structure and how the arrival of phosphorus causes the upward slope. The good fit of this simulated fine structure to the experimental data as shown in Figure 2.13 will also hold for the rest of the interference oscillations. This is illustrated in Figure 2.11, where the reflectance curves and reflectance derivatives match (and the fine structure amplitudes and turning points agree as well). The closeness of the fit and the correlation of the significant features discussed above support the ROSK model of the growth process and its effects on the reflectance measurements.

An important aspect of the behavior of the SRL kinetics which can be seen in Figure 2.13 is the difference between a phosphorus-terminated and gallium-terminated surface at the end of a cycle sequence. We had originally expected a phosphorus-terminated surface at the end of each 3 s cycle, where the TEG pulse is almost completely used up through desorption or formation of GaP, leaving some phosphorus in the SRL at the start of the next pulse cycle. However, simulated reflectance data with this type of behavior could not fit the experimental data. Instead, a set of parameters which resulted in a gallium-terminated surface (where the TEG pulse is not all used up at the end of the cycle time, leaving an amount of gallium in the SRL being carried over to the next cycle) gave a much more accurate fit as described above.

The data sets measured at the second angle (PR70) have structures and features similar to the PR75 data, with the major difference being the inversion of interference oscillation maxima/minima since the angles are on opposite sides of the pseudo-Brewster angle. Analysis of these measurements using the same model results in parameters similar to those found for PR75 (which are given below) and a similar fit of the reflectance data. There are a few noticeable differences between the two, which can be explained by the measurements being taken with light beams hitting different points on the surface. If the growth is somewhat uneven this could cause differences in the parameters in the growth model when the two data sets are compared.

The values used in the calculations of the model are as follows. The molar volumes used in the minimization process are $V_{TBP} = 128.6$ cm^3/mol, $V_{TEG} = 148$ cm^3/mol, $V_1 = 17$ cm^3/mol, $V_2 = 13$ cm^3/mol, $V_3 = 11.8$ cm^3/mol, and $V_{GaP} = 12.2$ cm^3/mol. The sticking coefficients used are $\beta_{TBP} = 0.15$ and $\beta_{TEG} = 1.0$, the effective SRL thickness parameter is $\alpha_{SRL} = 0.75$, and the substrate is a 2-inch diameter circular wafer. The geometrical parameter $\gamma = 0.025$ was estimated in the minimization process.

The numerical simulations were implemented using programs written in MATLAB code. The differential equations were solved numerically by the built-in function "*ode23*", an adaptive mesh and low order Runge-Kutta method. The optimization problems were solved using either a Nelder-Mead algorithm [29, 30, 31] or a Hooke-Jeeves procedure [29]. We are grateful to Professor C. T. Kelley for providing us with the code "*nelder*" implementing the Nelder-Mead algorithm and to Mr. D. Bortz for providing us with the code "*hj*" implementing the Hooke-Jeeves procedure. Both Professor Kelley and Mr. Bortz are in the Department of Mathematics at N.C. State.

We estimated the following parameters by averaging the results of independent best fits of the experimental data sets with varied TEG positions and TEG flow rates. The preliminary three-layer

stack problem resulted in $\epsilon_3 = 10.60 - 0.06i$ and $\epsilon_4 = 15.82 - 0.27i$. With these, the parameter estimation using the four-layer stack model gave the following parameter values: rate constants of $k_1 = 3.31$ s^{-1}, $k_2 = 1.55$ s^{-1}, $k_3 = 2.14$ s^{-1}, $k_4 = 0.052$ s^{-1}, and $k_{GaP} = 2.0$ s^{-1}, and optical responses of $F_1 = 13.46 - 0.13i$, $F_2 = 13.56 - 0.0i$ and $F_3 = 19.36 - 11.02i$. An earlier study on the decomposition kinetics of TEG, analyzing the PR responses to TEG exposure after growth interruptions [17], gave rate constants of $k_3 = 0.4$ s^{-1} and $k_{GaP} = 0.24$ s^{-1}. The higher value for k_{GaP} here was expected since no thermally activated hydrogen was employed to the growth surface and the flow rates were lower by a factor of about 2 in our experiments.

2.6 Conclusions

In this chapter we introduced a reduced order surface kinetics model using generalized reaction rate parameters to describe the decomposition kinetics of the organometallic precursor species TBP and TEG in heteroepitaxial growth of a gallium phosphide film on a silicon substrate. The set of coupled differential equations that describe the surface reaction kinetics provide information about the dynamics of molar concentrations of precursor fragments stored in the surface reaction layer and their incorporation into the underlying growing film. We fitted sets of low-pressure PCBE experimental data using the ROSK model to identify the unknown parameters involved in the surface kinetics and study their effects on the PR measurements. The results showed that the mathematical model can be used to effectively represent the deposition process and predict the large-scale and small-scale features of the experimental data. A true experimental validation of the ROSK model's predicted surface reaction layer constituents and their concentrations will require the development of highly surface sensitive, molecular specific diagnostic techniques that allow analysis of the dynamics in the SRL under steady-state growth. For this, the application of PRS in the infrared wavelength regime, using tuneable laser sources, has been proposed. To extend the chemical vapor deposition model to the high-pressure case, we will need to add the gas-phase species transport model discussed in the next chapter to the ROSK model for surface processes.

Chapter 3

Reduced Order Modeling of Gas-Phase Species Transport

3.1 Introduction

This chapter will present the gas-phase flow model of the high-pressure chemical vapor deposition reactor, largely based on the previous work by other members of the N.C. State research group in [6, 7]. These papers construct mathematical models with which to simulate and control the flow of precursor species from the inlet to the substrate surface. We will be coupling a version of this gas-phase model, described below, with the reduced order surface kinetics model discussed in Chapter 2 to create a model representing the whole HPCVD process from inlet flow to reactions causing film growth (this linkage is done in Section 6.2). The species transport process dynamics are given by the partial differential equations for continuity, momentum, energy and species balances, including multiple species and gas-phase reactions [32, 33, 34, 35]. The formulation is quasi-steady, with steady-state continuity, momentum and energy equations, and transient species equations. These general equations representing the reactor transport dynamics are discussed in Section 3.2. A generic numerical simulation of the time-dependent species equations using a standard finite element, finite difference or spectral method would lead to a system of ordinary differential equations far too large to be used for model-based real-time feedback control. Thus we use instead the proper orthogonal decomposition (POD) reduced basis method to limit the number of equations necessary, making the implementation of model-based feedback control possible. The POD method finds a set of basis functions which incorporate the dynamics of the particular problem (here the HPCVD species transport) and thus can represent the problem using very few basis functions compared with more

general methods. The POD itself and its properties are described in Section 3.3, while the use of the POD basis in a Galerkin formulation to obtain the reduced order ODE system is the subject of Section 3.4. Section 3.5 gives some overall conclusions on the gas-phase modeling process.

3.2 Transport Equations

The gas dynamics equations are considered for a case with only trace amounts of the precursors mixed with the carrier gas (N_2). With this dilute assumption the continuity, momentum and energy equations can be solved as steady-state equations, based on the properties of the dominant carrier gas and independent of the reactant concentrations:

$$\vec{\nabla} \cdot (\rho \vec{v}) = 0 \tag{3.1}$$

$$\rho \vec{v} \cdot \vec{\nabla} \vec{v} = -\vec{\nabla} P + \vec{\nabla} \cdot \vec{\tau} - \rho \vec{g} \tag{3.2}$$

$$\rho c_p \vec{v} \cdot \vec{\nabla} T = \vec{\nabla} \cdot (k \vec{\nabla} T), \tag{3.3}$$

where the viscous stress tensor is given by

$$\vec{\tau} = -\frac{2}{3}\mu(\vec{\nabla} \cdot \vec{v})\vec{I} + \mu(\vec{\nabla}\vec{v} + \vec{\nabla}\vec{v}^T). \tag{3.4}$$

Here $\vec{v}$, T and P are the velocity, temperature and pressure, $\vec{g}$ is the gravitational acceleration, and μ, c_p and k are the viscosity, specific heat and conductivity of the carrier gas. The density variation is modeled as

$$\rho = \rho_0[1 - \beta(T - T_0)], \tag{3.5}$$

with a reference temperature T_0, a reference density ρ_0 calculated from the ideal gas law at the reference temperature and reactor pressure, and the volume coefficient of expansion $\beta = 1/T$.

We will consider a three-dimensional rectangular-box-shaped domain describing the reactor depicted in Figure 3.1. Previous work [6, 7] used a two-dimensional representation of an earlier HPCVD reactor, but all three dimensions are necessary for this model, because in addition to the predominantly lengthwise gas flow and vertical deposition there is now also a horizontal measurement of optical absorption across the width of the reactor (to be discussed in Section 6.3). Our numerical model will represent an area of this reactor that is 150 mm long, 50 mm wide, and only 1 mm high. A side-view cross-section of this is shown in Figure 3.2, with the very small height not to scale. The portions of the reactor outside this domain, specifically the narrow inflow and outflow regions and the part of the box-shaped reactor near the inflow (seen in Figure 3.1) are not included in the model.

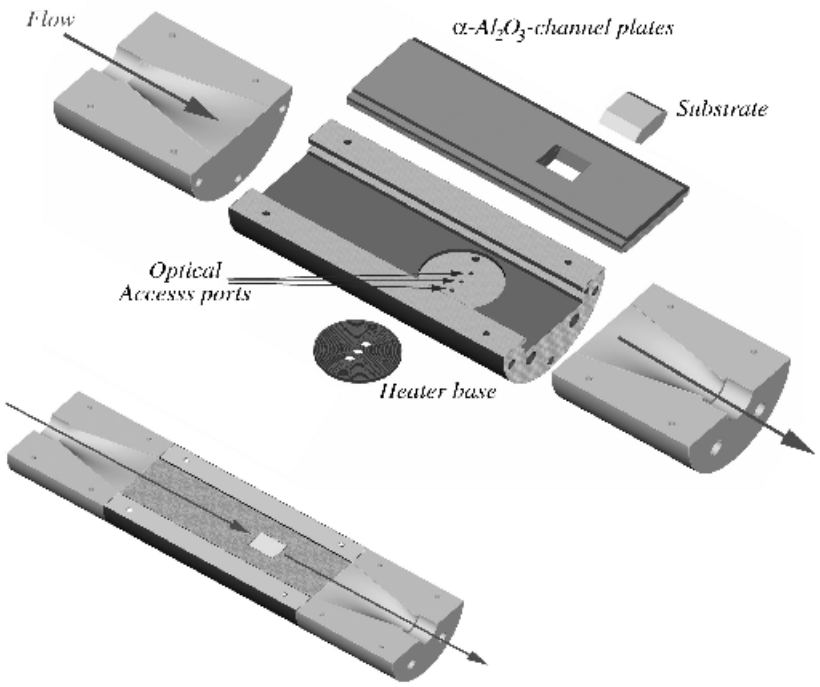

Figure 3.1: Three-dimensional view of the HPCVD reactor.

There are two substrates on which the growth is to take place, each 25 mm long by 20 mm wide and placed in the center of the reactor model (width-wise and length-wise), located symmetrically on both the top and bottom of the flow channel. The growth is driven thermally by a circular heating element, or susceptor, 40 mm in diameter and centered behind each substrate. The substrate and heating element placement can be seen in Figures 3.1 and 3.2, as well as in the photograph of the physical reactor flow channel in Figure 1.2. The optical access ports noted in Figure 3.1 in the heater beneath the substrate are for a variation on the PRS monitoring technique. This will be used in the physical reactor as a second measurement of the system, but our model will only involve the optical absorption measurement described in Section 6.3. The gas flow enters through the inlet seen at the left end of the reactor in these figures, flows across the substrate surfaces (depositing some of the gallium species there), and exits the outlet at the right end of the reactor. Here we consider Nitrogen (N_2) as the carrier gas, at a pressure of 10 atmospheres. Temperature-dependent values of μ, c_p and k for N_2 are linearly interpolated from values in the literature [36, 37, 38].

The boundary conditions for the steady-state flow model will be formulated as follows. There

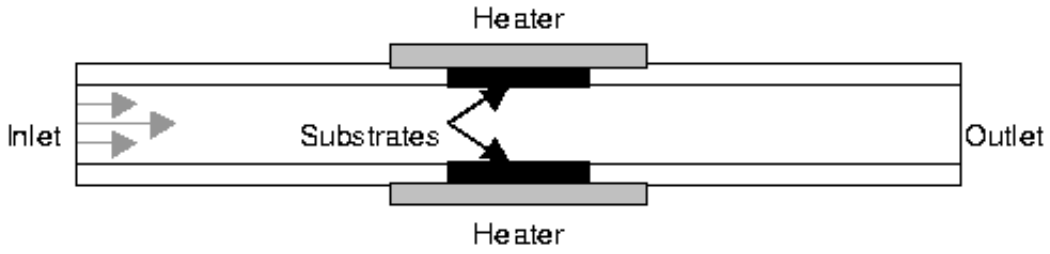

Figure 3.2: HPCVD reactor side-view cross-section (not to scale).

is a no-slip (zero velocity) condition on all walls of the reactor, as well as the substrate, while the inflow velocity profile corresponds to rectangular duct flow [39] with a flow rate of 10 standard liters per minute (slpm). The temperature boundary condition is room temperature (298 K) at the inlet and 1000 K at the heaters. Boundary conditions on the temperature at the rest of the reactor walls include conduction from the heaters along the walls, radiation off of the surfaces, convective heat loss to the environment, and modeled heat loss to the outside of the reactor. A special boundary condition in the FIDAP package, described in [34], page 6-14, is used to obtain a smooth outflow condition. The pressure is involved in equations (3.1)-(3.5) only as $\vec{\nabla}P$, so therefore only pressure differences matter, not absolute values, and an implicit $P = 0$ boundary condition is used at the outlet. A Galerkin finite element method with weighted residuals for the degrees of freedom ($\vec{v}$, T and P) is used to discretize the gas flow equations. The discretization uses a mixed formulation with 4983 brick elements (corresponding to 47131 nodes), with piecewise linear discontinuous elements for pressure and quadratic (27-noded) elements for the other degrees of freedom. The commercial software package FIDAP was used to solve the finite element problem for values of $\vec{v}$, T and P at the nodal points.

The solutions for velocity, temperature and density that are found from equations (3.1)-(3.5) can then be used in the solution of the time-dependent species equations for the precursor mass fractions,

$$\frac{\partial Y_n}{\partial t} + \vec{v} \cdot \vec{\nabla} Y_n = \frac{1}{\rho} \vec{\nabla} \cdot (\rho D_n \vec{\nabla} Y_n) + \sum_{i=1}^{N_R} r_{ni}, \tag{3.6}$$

where Y_n is the mass fraction of the nth species, D_n is the diffusivity of species n, N_R is the number of gas phase reactions, and r_{ni} is the rate of production of species n in the ith reaction.

In the HPCVD experiments to follow we will consider trimethylgallium, or TMG ($Ga(CH_3)_3$),

and phosphine (PH_3) as source materials for the growth of the III-V film gallium phosphide (GaP). For now we only look at the TMG pulse, in terms of representing it by POD modes and using those to control its contribution to the film deposition. This can be done because of the pulsing of the inputs with pauses in between, so that the gallium species and phosphorus species are not present in the reactor at the same time (though a constant overall gas flow profile is maintained throughout the pulsing cycle by the carrier gas). This pulsing prevents nucleation of the film in the gas phase. For the gallium transport we will consider three gas-phase species (aside from the carrier gas): Y_1 representing the mass fraction of TMG, Y_2 that of dimethylgallium (DMG), and Y_3 that of monomethylgallium (MMG).

We will assume that there are $N_R = 2$ significant gas phase reaction mechanisms for the gallium species. There is the reaction for the decomposition of TMG to DMG and a methyl molecule, $Ga(CH_3)_3 \rightarrow Ga(CH_3)_2 + CH_3$, and the decomposition of DMG to MMG and a methyl molecule, $Ga(CH_3)_2 \rightarrow GaCH_3 + CH_3$. These decompositions can be described as first-order Arrhenius reactions with rates of production given by

$$r_{ni} = \nu_{ni} \frac{W_n}{W_{m_i}} k_i e^{-E_i/RT} Y_{m_i}, \tag{3.7}$$

where m_i is the number of the species which is the source in reaction i. The parameter ν_{ni} is the stoichiometric constant for species n in reaction i, W_n and W_{m_i} are the molecular weights of those particular species, and k_i is the rate constant and E_i the activation energy for reaction i. The values of ν_{ni}, k_i and E_i for the two reactions we consider are taken from [40, 41] and are given in Table 3.1. The molar weights for the three species are $W_{TMG} = 114.8$ g/mol, $W_{DMG} = 99.79$ g/mol, and

Reaction i	Species n	$\nu_{ni} k_i$ (s^{-1})	E_i (kcal/mol)
1	1 (m_1)	-5.5×10^{15}	61.0
1	2	5.5×10^{15}	61.0
2	2 (m_2)	-8.7×10^{7}	35.4
2	3	8.7×10^{7}	35.4

Table 3.1: Rate constants and activation energies for gas-phase reactions.

$W_{MMG} = 84.755$ g/mol, and $R = 1.99$ cal/(mol·K) is the universal gas constant. The temperature dependent values of the diffusivities D_n are linearly interpolated from values in the literature [42]. The methyl molecules do not participate in the film growth and, due to the dilute approximation, do not contribute to the transport properties, so we do not include them in the transport equations.

The boundary conditions for the species transport equations will be as follows. We consider the substrate, as well as the parts of the reactor walls directly above the heated susceptor, to be perfectly absorbing, and the other walls completely non-absorbing, for each species. At the inlet, the source species (TMG) is set to desired values, and will be used as the input for control of the growth process, while the other two species are set to 0. Zero normal flux is assumed at the outflow, so that $\partial Y_n/\partial \overrightarrow{n} = 0$. The initial condition of the system is that no species are present in the reactor. Thus the entire time-dependent species transport system based on equation (3.6) is given by

$$\begin{aligned}
\frac{\partial Y_n}{\partial t} + \overrightarrow{v} \cdot \overrightarrow{\nabla} Y_n &= \frac{1}{\rho} \overrightarrow{\nabla} \cdot (\rho D_n \overrightarrow{\nabla} Y_n) + \sum_{i=1}^{N_R} r_{ni} \\
Y_n(0, \overrightarrow{x}) &= 0 \\
Y_1(t, \overrightarrow{x}) &= u_1 \quad \text{at inflow} \\
Y_n(t, \overrightarrow{x}) &= 0 \quad \text{at inflow } (n = 2, 3) \\
Y_n(t, \overrightarrow{x}) &= 0 \quad \text{at substrate, susceptor} \\
\frac{\partial Y_n(t, \overrightarrow{x})}{\partial \overrightarrow{n}} &= 0 \quad \text{at walls, outflow,}
\end{aligned} \tag{3.8}$$

where r_{ni} are the reaction production rates from equation (3.7) and u_1 is the control input which will be discussed later.

We use a penalty boundary formulation on the species transport problem in order to change all the boundary conditions to Neumann conditions. This prepares the system for the Galerkin procedure to be discussed in Section 3.4 and the control problem to be described in Chapter 6. Under the penalty boundary formulation the system (3.8) is modified to become

$$\begin{aligned}
\frac{\partial Y_n}{\partial t} + \overrightarrow{v} \cdot \overrightarrow{\nabla} Y_n &= \frac{1}{\rho} \overrightarrow{\nabla} \cdot (\rho D_n \overrightarrow{\nabla} Y_n) + \sum_{i=1}^{N_R} r_{ni} \\
Y_n(0, \overrightarrow{x}) &= 0 \\
\frac{\partial Y_1(t, \overrightarrow{x})}{\partial \overrightarrow{n}} &= \frac{1}{\epsilon} \left(Y_1(t, \overrightarrow{x}) - u_1 \right) \quad \text{at inflow} \\
\frac{\partial Y_n(t, \overrightarrow{x})}{\partial \overrightarrow{n}} &= \frac{1}{\epsilon} Y_n(t, \overrightarrow{x}) \quad \text{at inflow } (n = 2, 3) \\
\frac{\partial Y_n(t, \overrightarrow{x})}{\partial \overrightarrow{n}} &= \frac{1}{\epsilon} Y_n(t, \overrightarrow{x}) \quad \text{at substrate, susceptor} \\
\frac{\partial Y_n(t, \overrightarrow{x})}{\partial \overrightarrow{n}} &= 0 \quad \text{at walls, outflow,}
\end{aligned} \tag{3.9}$$

where ϵ is a small parameter (for our simulations we use $\epsilon = 10^{-3}$). With sufficient regularity assumed, it can be argued that as $\epsilon \to 0$ the solution of (3.9) approximates the solution to the

problem in (3.8) with $Y_1(t, \overrightarrow{x}) = u_1$, $Y_{2,3}(t, \overrightarrow{x}) = 0$ at the inlet and $Y_n(t, \overrightarrow{x}) = 0$ at the substrate (see [43, 44] for related discussions). This penalty formulation was successfully implemented in the previous feedback controlled HPCVD simulations in [6, 7].

3.3 The Proper Orthogonal Decomposition

The Proper Orthogonal Decomposition (POD), also known as the Karhunen-Loève expansion [45, 46] or principal component analysis [47], is a method of choosing a set of basis functions tailored to a particular problem so that a minimal number of functions are needed to represent that problem's dynamics. This is a well-known method of feature extraction in the pattern recognition field [48], and it has been used on many types of physical problems, including turbulent flows [49, 50, 51, 52, 53]. It has been used in open-loop control of CVD systems [5, 54], and recently in closed-loop control of CVD systems by members of this N.C. State research group [6, 7], for discretizing species transport PDEs like the system (3.9) so that they may be numerically solved more quickly. A standard finite difference, finite element, or spectral method of discretization will result in a very large system of ordinary differential equations. The POD method uses observations of the problem dynamics to choose basis functions which include as much of those dynamics as possible, thus making it possible to reduce the number of ODEs needed from hundreds or thousands down to the order of ten or less. The most important consequence of this in our work is that the POD system is small enough to be used to generate a real-time feedback control, whereas the system resulting from a generic discretization method would require much too long to solve.

In our problem we use FIDAP simulations of species flow as described in Section 3.2 to obtain a set of experimental observations, or "snapshots", of the species mass fractions at various times, to serve as the raw material for finding the POD modes. The snapshot data set for the nth species consists of K vectors, $U_n = \{u_{n1}^N, u_{n2}^N, ..., u_{nK}^N\}$, each vector consisting of the N nodal values of the mass fraction of the nth species at a different time during the simulation. This simulation is done starting with zero mass fraction for each species in the body of the reactor, a constant TMG mass fraction input serving as a nontrivial "control" for 0.5 s, followed by a clearing pulse of zero input for another 0.5 s. The simulation uses a backward Euler method with fixed time steps of 0.01 s for the time integration. A total of 100 snapshots are obtained from the 0.01 s time intervals.

For the purposes of describing the process of finding the POD, we will think of the snapshots $\left\{u_{ni}^N\right\}$ not as length N vectors but as functions $\left\{u_{ni}^N(x)\right\}$ of the location x in the reactor domain Ω.

We also will eliminate the species subscript n and length-N superscript for now, so that the set of snapshots is given by $U(x) = \{u_1(x), u_2(x), ..., u_K(x)\}$, remembering that we are considering the snapshots for a single species. Following the description of [5], we look for the set of K POD modes that most closely match the snapshot data, so that the POD modes $z(x)$ maximize

$$\frac{1}{K}\sum_{i=1}^{K}\left|(u_i, z)\right|^2, \tag{3.10}$$

while still being independent of each other and having

$$(z, z) = \|z\|^2 = 1.$$

Here $(\cdot,\cdot)$ and $\|\cdot\|$ are the $\mathbf{L}^2$ inner product and norm. Specifically we look for POD modes that are linear combinations of the snapshots, so that

$$z(x) = \sum_{i=1}^{K} a_i u_i(x), \tag{3.11}$$

with the coefficients a_i to be found so that (3.10) is maximized.

To start, we define the function

$$\mathbf{K}(x, x') = \frac{1}{K}\sum_{i=1}^{K} u_i(x)u_i(x'), \tag{3.12}$$

and the operator $\mathbf{R}$ of z

$$\left(\mathbf{R}z\right)(x) = \int_\Omega \mathbf{K}(x, x')z(x')dx'.$$

By manipulating these defined equations we find that

$$\begin{aligned}
(\mathbf{R}z, z) &= \int_\Omega \mathbf{R}z(x)z(x)dx \\
&= \int_\Omega\int_\Omega \mathbf{K}(x, x')z(x')dx'z(x)dx \\
&= \frac{1}{K}\sum_{i=1}^{K}\int_\Omega\int_\Omega u_i(x)u_i(x')z(x')dx'z(x)dx \\
&= \frac{1}{K}\sum_{i=1}^{K}\left|(u_i, z)\right|^2.
\end{aligned}$$

It also follows that $(\mathbf{R}z_1, z_2) = (z_1, \mathbf{R}z_2)$ for any $z_1, z_2 \in \mathbf{L}^2(\Omega)$, so therefore $\mathbf{R}$ is a nonnegative symmetric operator on $\mathbf{L}^2(\Omega)$. With this property, the problem of maximizing the expression (3.10) is the same as finding the largest eigenvalue in the problem

$$\mathbf{R}z = \lambda z \tag{3.13}$$

subject to $\|z\|^2 = 1$, or equivalently,

$$\int_\Omega \mathbf{K}(x,x')z(x')dx' = \lambda z(x) \tag{3.14}$$

with $\|z\|^2 = 1$.

By substituting the definitions of z and $\mathbf{K}$ in equations (3.11) and (3.12) into equation (3.14), we obtain

$$\sum_{i=1}^{K}\left[\sum_{j=1}^{K}\frac{1}{K}\int_\Omega u_i(x')u_j(x')dx'a_j\right]u_i(x) = \sum_{i=1}^{K}\lambda a_i u_i(x),$$

which can be rewritten as the matrix eigenvalue problem

$$C\psi = \lambda\psi,$$

where the matrix and eigenvector are given by

$$\begin{aligned} C_{ij} &= \frac{1}{K}\int_\Omega u_i(x')u_j(x')dx', \\ \psi &= \left[a_1, a_2, ..., a_K\right]^T. \end{aligned}$$

C is a nonnegative Hermitian matrix, so therefore it has a complete set of orthogonal eigenvectors $\{\psi_1, \psi_2, ..., \psi_K\}$, with $\psi_k = \left[a_1^k, a_2^k, ..., a_K^k\right]^T$, ordered from largest to smallest eigenvalues. The expression (3.10) is then maximized by the POD mode obtained from the eigenvector corresponding to the largest eigenvalue by setting

$$z_1(x) = \sum_{i=1}^{K} a_i^1 u_i(x) = U(x)\psi_1.$$

The other POD modes are calculated similarly from the remaining $K-1$ eigenvectors.

In the case of the HPCVD problem, the snapshots are actually given in terms of length N vectors of the finite element coefficients, so instead of the integral version of the correlation matrix C, it will be given as $C_{ij} = (1/K)u_i^T u_j$. For a particular species n, the K POD modes Z_n are therefore obtained by taking the linear combination $Z_n = \{z_{n1}^N, z_{n2}^N, ..., z_{nK}^N\} = U_n\Psi_n$ in terms of the snapshots $U_n = \{u_{n1}^N, u_{n2}^N, ..., u_{nK}^N\}$ of that species. The columns of $\Psi_n = \{\psi_{n1}^K, \psi_{n2}^K, ..., \psi_{nK}^K\}$ are the orthogonal eigenvectors of the correlation matrix of the snapshots, given by the solution of $\left[(1/K)U_n^T U_n\right]\psi_{ni}^K = \lambda_{ni}\psi_{ni}^K$, ranked in descending order of the eigenvalues λ_{ni}. The POD basis elements are found from these POD modes by using generic finite element interpolation functions, such as the quadratic functions which will be used in Section 3.4. A Galerkin procedure on the

weak form of the original PDE system using a chosen number of these basis elements will result in an ODE system similar to that of a finite element Galerkin procedure but with far fewer equations.

The POD basis has certain desirable properties which result from the conditions of its creation. First, the POD modes can be shown to be orthonormal, as proved in [5]. In showing this and the other properties which will follow, we will use the notation from the earlier description of the POD procedure, removing the species n subscript and length-N superscript and considering the snapshots and POD modes as functions of x. The orthonormality property of the POD modes $\{z_k\}$ is based on the orthogonality of the eigenvectors $\{\psi_k\}$, and the condition that the eigenvector sizes are chosen so that

$$\psi_k \cdot \psi_n = \begin{cases} 1/(K\lambda_n) & k = n \\ 0 & k \neq n. \end{cases}$$

Using this, the inner product of two POD modes is given by

$$\begin{aligned}
(z_k, z_n) &= \int_\Omega z_k(x) z_n(x) dx \\
&= \int_\Omega \sum_{i=1}^{K} a_i^k \psi_i(x) \sum_{j=1}^{K} a_j^n \psi_j(x) dx \\
&= \sum_{i=1}^{K} a_i^k K \sum_{j=1}^{K} \left[\frac{1}{K} \int_\Omega \psi_i(x) \psi_j(x) dx \right] a_j^n \\
&= K \sum_{i=1}^{K} a_i^k \sum_{j=1}^{K} C_{ij} a_j^n \\
&= K \psi_k \cdot (C \psi_n) \\
&= K \psi_k \cdot (\lambda_n \psi_n) \\
&= K \lambda_n \psi_k \cdot \psi_n \\
&= \begin{cases} 1 & k = n \\ 0 & k \neq n. \end{cases}
\end{aligned}$$

Thus the POD modes are orthonormal.

The second property we will show is that the POD coefficients are uncorrelated. This can be proved in the following manner (expanded from the proof in [50], page 77, or [51], page 237). Consider a particular snapshot u_i to be represented in terms of the POD modes as $u_i(x) = \sum_{j=1}^{K} b_{ij} z_j(x)$. We then wish to show that $(1/K) \sum_{i=1}^{K} b_{ij} \cdot b_{ik} = \lambda_j \delta_{jk}$. We begin with the definition of $\mathbf{K}(x, x')$, rewritten in terms of these snapshot representations as

$$\begin{aligned}
\mathbf{K}(x,x') &= \frac{1}{K}\sum_{i=1}^{K} u_i(x)u_i(x') \\
&= \frac{1}{K}\sum_{i=1}^{K}\left[\sum_{j=1}^{K} b_{ij}z_j(x)\sum_{k=1}^{K} b_{ik}z_k(x')\right] \\
&= \sum_{j=1}^{K}\sum_{k=1}^{K}\left[\frac{1}{K}\sum_{i=1}^{K} b_{ij}b_{ik}\right] z_j(x)z_k(x'). \qquad (3.15)
\end{aligned}$$

Leaving this formula temporarily, any eigenvalue λ_k and corresponding POD mode $z_k(x)$ satisfy the eigenvalue problem for $\mathbf{R}$ in equation (3.13). Manipulating this and using the orthonormality property of $\{z_k\}$ will lead to

$$\begin{aligned}
\left(\mathbf{R}z_k\right)(x) &= \lambda_k z_k(x) \\
&= \sum_{j=1}^{K}\lambda_j z_j(x)\delta_{jk} \\
&= \sum_{j=1}^{K}\lambda_j z_j(x)\int_\Omega z_j(x')z_k(x')dx' \\
&= \int_\Omega\left[\sum_{j=1}^{K}\lambda_j z_j(x)z_j(x')\right] z_k(x')dx'.
\end{aligned}$$

However, by definition we also have

$$\left(\mathbf{R}z_k\right)(x) = \int_\Omega \mathbf{K}(x,x')z_k(x')dx'.$$

Both of these expressions hold for each POD mode z_k, and, since as $K\to\infty$ every function of x can be written as $f(x)=\sum_{k=1}^{K} z_k(x)$, both $\mathbf{R}$ expressions above hold for every function $f(x)$ as well. This means that the interiors of the two can be equated, so that

$$\mathbf{K}(x,x') = \sum_{j=1}^{K}\lambda_j z_j(x)z_j(x'). \qquad (3.16)$$

Now we can equate in turn the values for $\mathbf{K}(x,x')$ in equations (3.15) and (3.16), to obtain

$$\sum_{j=1}^{K}\sum_{k=1}^{K}\left[\frac{1}{K}\sum_{i=1}^{K} b_{ij}b_{ik}\right] z_j(x)z_k(x') = \sum_{j=1}^{K}\lambda_j z_j(x)z_j(x'). \qquad (3.17)$$

Note also that $\{z_j(x)z_k(x')\}$ are an orthonormal family of functions in $\mathbf{L}^2(\Omega\times\Omega)$, since

$$\begin{aligned}\int_\Omega \int_\Omega [z_j(x)z_k(x')]\,[z_i(x)z_n(x')]\,dxdx' &= \int_\Omega [z_j(x)z_i(x)]\,dx \int_\Omega [z_k(x')z_n(x')]\,dx' \\ &= \delta_{ij}\delta_{kn} \\ &= \begin{cases} 1 & i=j \text{ and } k=n \\ 0 & i\neq j \text{ or } k\neq n. \end{cases}\end{aligned}$$

This means that equation (3.17) reduces down to

$$\frac{1}{K}\sum_{i=1}^{K} b_{ij}b_{ik} = \lambda_j \delta_{jk}$$

for each pair (j,k). Therefore the POD coefficients $\{b_{ik}\}$ are uncorrelated.

The final property of the POD which we will look at is the fact that the representation of the original data set $U = \{u_i(x)\}$ in terms of the first M POD modes,

$$\{u_i(x)\} = \left\{ \sum_{j=1}^{M} b_{ij} z_j(x) \right\},$$

for any $M < K$, maximizes the "energy" of such an M-mode reduced basis representation of the snapshots. That is, for a representation in terms of any other orthonormal basis,

$$\{u_i(x)\} = \left\{ \sum_{j=1}^{M} c_{ij} \widetilde{z}_j(x) \right\},$$

the POD coefficients will contain more "energy", as measured by $\sum_{i=1}^{M} \langle b_i^2 \rangle$ with the notation $\langle b_i b_j \rangle = (1/K)\sum_{k=1}^{K} b_{ki}b_{kj}$, than the coefficients for the other basis. We will prove that

$$\sum_{i=1}^{M} \langle b_i^2 \rangle = \sum_{i=1}^{M} \lambda_i \geq \sum_{i=1}^{M} \langle c_i^2 \rangle,$$

for the POD basis and other basis as written above. The proof is an expansion of that given in [50, 51] with the uncorrelated-coefficients proof given above. The "energy" described by Berkooz in these places is also referred to elsewhere as the data variability.

We already have $\sum_{i=1}^{M} \langle b_i^2 \rangle = \sum_{i=1}^{M} \lambda_i$, from the last proof. Now, in terms of the other orthonormal basis $\{\widetilde{z}_j\}$, assumed to be completed to form an infinite-dimensional basis, we have

$$\begin{aligned}\mathbf{K}(x,x') &= \frac{1}{K}\sum_{k=1}^{K}\left[\sum_{i=1}^{\infty} c_{ki}\widetilde{z}_i(x)\sum_{j=1}^{\infty} c_{kj}\widetilde{z}_j(x')\right] \\ &= \sum_{i=1}^{\infty}\sum_{j=1}^{\infty} \langle c_i c_j \rangle\, \widetilde{z}_i(x)\widetilde{z}_j(x').\end{aligned}$$

Using this and looking at the operator $\mathbf{R}f$ for a function $f(x) = \sum_{k=1}^{\infty} d_k \widetilde{z}_k(x)$ in terms of the new basis, we have

$$\begin{aligned}
(\mathbf{R}f)(x) &= \int_\Omega \left[\sum_{i=1}^{\infty}\sum_{j=1}^{\infty} \langle c_i c_j\rangle \widetilde{z}_i(x)\widetilde{z}_j(x')\right]\left[\sum_{k=1}^{\infty} d_k \widetilde{z}_k(x')\right] dx' \\
&= \sum_{i=1}^{\infty}\sum_{j=1}^{\infty}\sum_{k=1}^{\infty} d_k \langle c_i c_j\rangle \widetilde{z}_i(x) \int_\Omega \widetilde{z}_j(x')\widetilde{z}_k(x')dx' \\
&= \sum_{i=1}^{\infty}\sum_{j=1}^{\infty}\sum_{k=1}^{\infty} d_k \langle c_i c_j\rangle \widetilde{z}_i(x)\delta_{jk} \\
&= \sum_{i=1}^{\infty}\sum_{j=1}^{\infty} d_j \langle c_i c_j\rangle \widetilde{z}_i(x).
\end{aligned}$$

Here $\mathbf{R}$ is mapping the function, in the $\{\widetilde{z}_j\}$ basis representation, given by

$$\begin{bmatrix} \widetilde{z}_1(x) & \widetilde{z}_2(x) & \cdots \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \\ \vdots \end{bmatrix},$$

to the new function

$$\begin{bmatrix} \widetilde{z}_1(x) & \widetilde{z}_2(x) & \cdots \end{bmatrix} \begin{bmatrix} \langle c_1c_1\rangle & \langle c_1c_2\rangle & \cdots \\ \langle c_2c_1\rangle & \langle c_2c_2\rangle & \\ \vdots & & \ddots \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \\ \vdots \end{bmatrix}.$$

We call the matrix of $\langle c_i c_j\rangle$ elements $\widehat{C}$; this is a matrix representation of $\mathbf{R}$, mapping the old basis coefficients $\{d_i\}$ to the new coefficients $\{\sum_j d_j \langle c_i c_j\rangle\}$. The matrix

$$\widehat{Q} = \begin{bmatrix} I_M & 0 & \cdots \\ 0 & 0 & \\ \vdots & & \ddots \end{bmatrix},$$

containing an M-dimensional identity matrix I_M, restricts a vector or a matrix such as $\widehat{C}$, written in terms of the infinite-dimensional basis $\{\widetilde{z}_j\}$, by projecting it onto $\{\widetilde{z}_j\}_{i=1,\ldots,M}$. This results in

$$\widehat{C}\widehat{Q} = \begin{bmatrix} \widehat{C}_{M\times M} & 0 & \cdots \\ 0 & 0 & \\ \vdots & & \ddots \end{bmatrix},$$

where $\widehat{C}_{M\times M}$ is the upper left $M \times M$ block of $\widehat{C}$. Then a result from [55] or [56] is used which states that for this matrix representation $\widehat{C}$ of the operator $\mathbf{R}$, and this restriction matrix $\widehat{Q}$, that

$$\sum_{i=1}^{M} \lambda_i \geq tr\left(\widehat{C}\widehat{Q}\right) = \sum_{i=1}^{M} \left\langle c_i^2 \right\rangle .$$

This completes the second part of the proof, thus showing that the M-mode POD basis contains more of the data variability of the original snapshots in its basis coefficients than any other M-mode orthonormal basis.

As a consequence of the property described above, the amount of variability contained in a certain POD mode z_k is given by the eigenvalue λ_k, and the percentage of the total snapshot set variability in that mode is given by the ratio of that eigenvalue to the total of all eigenvalues, $\lambda_k / \sum_{j=1}^{K} \lambda_j$. The reason for ordering the POD modes from highest to lowest eigenvalues is to include as much of the variability of the system into the few first modes as possible. Therefore a POD reduced order system, when found in this way, can still be a very good representation of the dynamics of the system while using only a few modes (in fact it is the best representation using that number of modes, in the sense described above). Due to these characteristics, if desired the number M of POD modes to be used can be determined by taking enough that the data variability accounted for by them is more than a chosen amount, so that, for example,

$$\sum_{i=1}^{M} \lambda_i \Bigg/ \sum_{j=1}^{K} \lambda_j > 0.99.$$

In an analogous concept, it can be shown that the POD basis minimizes the mean square error of the representation over all choices of orthonormal bases [48].

3.4 Discretizing the Flow Problem

There are two different discretization formulas which we will apply to the flow problem (3.9). A finite element formula is used in the simulations done to obtain the snapshots for the POD mode generation described in the last section. For this purpose it plays the role of a reactor simulator, and results in a detailed system with a large number of basis functions. The POD modes are used in the second form of the discretization, in order to produce the reduced order gas-phase model. This uses only a small number of basis functions and will be used to find the real-time feedback control.

The finite element discretization is done in the standard way, starting with the weak formulation

of the species flow equation:

$$\int_\Omega \frac{\partial Y_n}{\partial t} w_j d\Omega = -\int_\Omega \left(\overrightarrow{v}\cdot\overrightarrow{\nabla} Y_n\right) w_j d\Omega - \int_\Omega D_n \overrightarrow{\nabla} Y_n \cdot \overrightarrow{\nabla} w_j d\Omega + \int_\Omega \frac{1}{\rho} D_n \left(\overrightarrow{\nabla} Y_n \cdot \overrightarrow{\nabla}\rho\right) w_j d\Omega + \sum_{i=1}^{N_R} \int_\Omega r_{ni} w_j d\Omega + \frac{1}{\epsilon}\int_{\Gamma 1,\Gamma 2} w_j D_n Y_n ds - \frac{1}{\epsilon}\int_{\Gamma 1} w_j D_n u_n ds, \tag{3.18}$$

for $n = 1, 2, 3$, with $u_n = 0$ for $n = 2, 3$ (the only species inlet flow we control is TMG). The important portions of the boundary involved in this formulation are $\Gamma 1$, which represents the inlet, and $\Gamma 2$, which represents the substrate and susceptor. The values of $\overrightarrow{v}$ and T, and by extension $\rho(T)$ and $D_n(T)$, are found from the steady-state flow simulations, and the reaction rates of production r_{ni} are found by equation (3.7) in terms of T and $\overrightarrow{Y}$. The standard finite element procedure here uses $N = 47131$ quadratic interpolation functions $\{\phi_k\}$ and nodal coefficients $\{y_{nk}\}$, so that the approximate species n mass fraction is

$$Y_n^N(t, \overrightarrow{x}) = \sum_{k=1}^{N} y_{nk}(t)\phi_k(\overrightarrow{x}). \tag{3.19}$$

The test functions are chosen as $w_j = \phi_j$, $j = 1, ..., N$ for each species. Substituting these test functions and equation (3.19) into the weak form (3.18), the problem reduces to a system of ordinary differential equations for the nodal coefficients,

$$\dot{y}^{3N}(t) = A^{3N} y^{3N}(t) + B^{3N} u(t), \tag{3.20}$$

where A^{3N} is a $3N \times 3N$ matrix, B^{3N} is a $3N \times 1$ vector and u is the one-dimensional control.

The POD discretization follows almost the same steps, with the preliminary problem of forming POD basis elements from the calculated POD modes $\{z_{nk}^N\}$ and the finite element quadratic interpolation functions $\{\phi_k\}$. The formula

$$\Phi_{nk}(\overrightarrow{x}) = \sum_{i=1}^{N} (z_{nk}^N)_i \phi_i(\overrightarrow{x})$$

gives a set of K basis functions $\{\Phi_{nk}\}$ for each species $n = 1, 2, 3$. Then the mass fraction of the nth species is approximated as a linear combination of the M_n most significant POD modes for that species,

$$Y_n^{M_n}(t, \overrightarrow{x}) = \sum_{k=1}^{M_n} y_{nk}(t)\Phi_{nk}(\overrightarrow{x}), \tag{3.21}$$

where $M_n << K << N$. Substitution of this approximation into (3.18), using test functions $w_j = \Phi_{nj}$ $(j = 1, ..., M_n)$ for the nth species, results in the reduced order system

$$\dot{y}^M(t) = A^M y^M(t) + B^M u(t), \tag{3.22}$$

where $M = \sum_{n=1}^{3} M_n$, A^M is an $M \times M$ matrix and B^M is an $M \times 1$ vector. This POD discretization produces an ODE system in terms of the POD coefficients $\{y_{nk}\}$, in a form appropriate for a feedback control implementation and of small enough order that real-time model-based control is possible.

3.5 Conclusions

The construction of the gas-phase model in this chapter has followed the work in [6, 7] on a similar reactor design. Although the main properties of the model are largely the same, there are differences in the species involved, the reactor geometry, and the parameters used. The first major difference is that the model we use here is three-dimensional, in contrast with the two-dimensional representations in the earlier work. This is necessary because of the second major difference between this model and the previous one: the real-time measurement of the current reactor will be done using an optical absorption measurement (see Section 6.3) which requires the modeling of the species mass fractions in the third dimension (across the width of the reactor). In contrast, the earlier problem assumed a measurement of the gallium flux at the substrate surface, which cannot be realistically acquired. The proper orthogonal decomposition (POD) algorithm is used here as in [6, 7] to find basis functions from snapshots of the full system dynamics and to obtain a reduced order system using them. With the penalty boundary formulation of the species equation and the Galerkin procedure to create a reduced order ODE system, the species transport model has been changed into a form such that the control methods to be discussed in Chapters 4 and 5 can be applied to it.

Chapter 4

Comparison of Feedback Control Methods for Nonlinear Dynamical Systems

4.1 Introduction

Having constructed a system modeling the high-pressure chemical vapor deposition dynamics in Chapters 2 and 3, we now need a methodology that will allow us to control this system. The most significant aspect of the problem is that it is nonlinear, due to the quadratic terms in the reduced order surface kinetics model (2.4)-(2.7). It is also fairly large, even using the reduced order models for both surface and gas phase dynamics, with a total of roughly 9-15 ordinary differential equations. Both the nonlinearity and the size of the system must be considered in finding a control method for this problem.

The optimal feedback control of a linear system is a subject which has been extensively studied (see e.g. [57] and the references therein). If the cost functional is quadratic in the state and control, and if we assume full state knowledge, then the optimal control is a linear state feedback law where the control gains are obtained by solving a differential/algebraic Riccati equation. The success of this linear quadratic regulator (LQR) problem is due to the successful development of robust and efficient algorithms for solving the Riccati equation. However, if the plant is described by nonlinear dynamics, then the optimal state feedback law is given in terms of the solution to the Hamilton-Jacobi-Bellman (HJB) equation [58]. The HJB equation provides the solution to the optimal control problem for general nonlinear systems; however, in most cases it is impossible to solve analytically. This has led to many methods being proposed in the literature for ways to obtain an approximate solution to the HJB equation as well as obtain a suboptimal feedback control for general nonlinear

dynamical systems.

One possibility is to construct the control as a power series, either by separating out the nonlinearities in the system into a power series, or by introducing a temporary variable and expanding around it. Then the first few terms in the series for the control are found by various techniques. This idea is based on considering the system as a perturbation of a linear system, with the control being an extension of the linear control (the first term of the power series is obtained by solving the Riccati equation for the solution of the linearized system). These types of methods are described in [59, 60, 61, 62].

Another approach is through successive approximation, where an iterative process is used to find a sequence of approximations approaching the solution of the HJB equation. This is done by solving a sequence of generalized Hamilton-Jacobi-Bellman (GHJB) equations and is discussed in [63, 64] in a general context. A more concrete technique for finding the desired solution is described in [65, 66], where a Galerkin procedure is used to find a numerical solution to the GHJB equation.

Other methods include the state-dependent Riccati equation (SDRE), which is an extension of the Riccati equation to nonlinear systems [67, 68, 69, 70, 71]. The coefficients in the SDRE are functions of the state instead of being constant-valued as in the linear case. This makes the equation much more difficult to solve. Also, in the nonlinear case the resulting control is only suboptimal. There is also a control method based on representing the curve of the system solution as a linear operator and then inverting it [72], as well as an exploration of the use of neural nets to solve the HJB equation [73].

Open-loop controls can also be found by a variety of different methods. While open-loop controls are generally not useful for practical applications, it is possible to extend them into closed-loop controls. This can be done with an interpolation over the state space as described in [74], for example. The open-loop control can be found by solving a two-point boundary value (TPBV) problem, which is obtained from the necessary optimality conditions. This can be solved by discretizing and then solving exactly [74], or else by one of several iterative methods (quasilinearization, gradient methods, etc.) as in [75, 76, 77, 78, 79].

It is also important to note that different methods are applicable to different problems. Some of the methods that we will investigate in detail can be used on a more general problem than the one that we will consider; for example, a problem with a nonlinear function of the control as well as the state. There are also methods which are designed for other specific problems, such as the bilinear system $\dot{x} = Ax + Bu + (\sum x_j N_j)u$. Methods to solve this type of problem are discussed in [80,

81, 82, 83, 84]. Another variation on the nonlinear control problem is an uncertain system, where unknown parameters are involved in the equation, as discussed in [85, 86]. A method dealing only with systems where the number of state and control variables are the same is presented in [87].

Which of these methods is the best choice for a particular problem may depend on the nature of the nonlinearities, the size of the system, whether the amount of control used or time needed for the method is a concern, and other factors. In this chapter a comprehensive comparison study of five of the methods mentioned above is performed with several test problems, so that we can draw some conclusions as to which methods seem best to use for certain types of problems, including our HPCVD problem. We begin by looking at the general nonlinear control problem statement in Section 4.2. The methods to be considered are described in Section 4.3. In particular, Section 4.3.3 describes the successive approximation method; because there was an error in one of the formulas in the original paper, this section contains a detailed derivation of the method. The results of the application of the methods to four examples are discussed in Section 4.4, and overall conclusions are given in Section 4.5.

4.2 Control Problem Statement

Consider a nonlinear system of the type

$$\begin{cases} \dot{x}(t) &= f(x(t)) + Bu(x(t)) \\ x(0) &= x_0, \end{cases} \tag{4.1}$$

where $f(x)$ is a nonlinear function of the state $x \in \Omega \subset R^m$, $u : \Omega \to R^k$ is the control, and B is a constant-valued $m \times k$ matrix. Also consider a cost functional

$$J(x_0, u) = \int_0^\infty \left(x^T Q x + u^T R u\right) dt$$

with a given constant-valued $m \times m$ symmetric positive semi-definite matrix Q and a $k \times k$ symmetric positive definite matrix R. The optimal control problem is to find a state feedback control $u^*(x)$ which minimizes the cost for all possible initial conditions x_0.

For the simpler linear problem, where $f(x) = A_0 x$, the optimal feedback control is given by $u^*(x) = -R^{-1}B^T\Pi x$, with Π solving the matrix Riccati equation $\Pi A_0 + A_0^T\Pi - \Pi B R^{-1} B^T \Pi + Q = 0$. The theories for this linear quadratic regulator (LQR) problem have been established for both the finite-dimensional and infinite-dimensional problems (see e.g. [88, 89]). In addition, stable and

robust algorithms for solving the Riccati equation have been developed and are well documented in many places in the literature and in textbooks.

For the nonlinear case, the optimal feedback control is known to be of the form

$$u^*(x) = -\frac{1}{2}R^{-1}B^T V_x(x),$$

where the function V is the solution to the Hamilton-Jacobi-Bellman equation

$$V_x^T(x)f(x) - \frac{1}{4}V_x^T(x)BR^{-1}B^T V_x(x) + x^T Qx = 0. \tag{4.2}$$

However, the HJB equation itself is very difficult to solve analytically for any but the simplest problems. Thus efforts have been made to numerically approximate the solution of the HJB equation, or to solve a related problem producing a suboptimal control, or to use some other process in order to obtain a usable feedback control. The following section will outline some such methods that have been proposed in the literature.

4.3 Feedback Control Methodologies for Nonlinear Systems

4.3.1 Power Series Approximation

One type of method of obtaining an approximate solution to the HJB equation (4.2) is by using a power series to approximate $V(x)$. This is done by Garrard and others in [59, 60, 61]. Consider the representation $V(x) = \sum_{n=0}^{\infty} V_n(x)$, where each $V_n(x) = O(x^{n+2})$. Also separate the nonlinear function $f(x)$ into powers of x as $f(x) = A_0x + \sum_{n=2}^{\infty} f_n(x)$ with $f_n(x) = O(x^n)$. Substituting these expansions into the HJB equation results in

$$\left(\sum_{n=0}^{\infty}(V_n)_x^T\right)\left(A_0x + \sum_{n=2}^{\infty} f_n(x)\right) - \frac{1}{4}\left(\sum_{n=0}^{\infty}(V_n)_x^T\right)BR^{-1}B^T\left(\sum_{n=0}^{\infty}(V_n)_x\right) + x^TQx = 0.$$

By separating out by powers of x, we obtain a series of equations,

$$(V_0)_x^T A_0x - \frac{1}{4}(V_0)_x^T BR^{-1}B^T(V_0)_x + x^TQx = 0, \tag{4.3}$$

$$(V_1)_x^T A_0x - \frac{1}{4}(V_1)_x^T BR^{-1}B^T(V_0)_x - \frac{1}{4}(V_0)_x^T BR^{-1}B^T(V_1)_x + (V_0)_x^T f_2(x) = 0, \tag{4.4}$$

$$(V_n)_x^T A_0x - \frac{1}{4}\sum_{k=0}^{n}\left[(V_k)_x^T BR^{-1}B^T(V_{n-k})_x\right] + \sum_{k=0}^{n-1}\left[(V_k)_x^T f_{n+1-k}(x)\right] = 0, \tag{4.5}$$

where $n = 2, 3, 4, ...$

Equation (4.3) can be solved with $V_0(x) = x^T \Pi x$, where the symmetric positive definite matrix Π solves the Riccati equation $\Pi A_0 + A_0^T \Pi - \Pi B R^{-1} B^T \Pi + Q = 0$. This gives the standard linear control. Equations (4.4) and (4.5) can be solved for V_n, $n = 1, 2, 3...$, by making V_n a scalar polynomial containing all possible combinations of products of the state elements with a total order of $n + 2$, and then substituting it into the equation and solving for the coefficients. This can be done for as many terms as desired, but in general this quickly gets very complicated, especially for problems with a large number of state variables. As a way of avoiding this, Garrard proposes a method of finding $(V_1)_x$ more simply, which results in a very easy method of finding a quadratic type control [59].

Instead of using the polynomial representation, we consider equation (4.4) as is and use the substitution $(V_0)_x = 2\Pi x$ (where Π is the solution to the Riccati equation) to obtain

$$(V_1)_x^T A_0 x - \frac{1}{4}(V_1)_x^T B R^{-1} B^T (2\Pi x) - \frac{1}{4}(2x^T \Pi) B R^{-1} B^T (V_1)_x + (2x^T \Pi) f_2(x) = 0.$$

Rearranging some terms, we find

$$x^T \left[A_0^T (V_1)_x - \Pi B R^{-1} B^T (V_1)_x + 2\Pi f_2(x) \right] = 0.$$

The quantity inside the brackets is zero when $(V_1)_x = -2(A_0^T - \Pi B R^{-1} B^T)^{-1} \Pi f_2(x)$. This along with the $(V_0)_x$ term gives a quadratic feedback control law of the form

$$u(x) = -R^{-1} B^T [\Pi x - (A_0^T - \Pi B R^{-1} B^T)^{-1} \Pi f_2(x)]. \tag{4.6}$$

If $f_2(x) = 0$ (that is, if there are no quadratic terms), then (4.4) is solved trivially by $V_1 = 0$. Equation (4.5) for $n = 2$ will then be of the form

$$(V_2)_x^T A_0 x - \frac{1}{4}(V_2)_x^T B R^{-1} B^T (V_0)_x - \frac{1}{4}(V_0)_x^T B R^{-1} B^T (V_2)_x + (V_0)_x^T f_3(x) = 0.$$

This is exactly the same form as equation (4.4), except that now V_2 is in the place of V_1 and f_3 is in the place of f_2. Thus the solution is analogous to that of equation (4.4), resulting in $(V_2)_x = -2(A_0^T - \Pi B R^{-1} B^T)^{-1} \Pi f_3(x)$ and a feedback control of the form

$$u(x) = -R^{-1} B^T [\Pi x - (A_0^T - \Pi B R^{-1} B^T)^{-1} \Pi f_3(x)].$$

If $f_3 = 0$ as well, then the resulting control will be the same except for f_4 in the place of f_3. In other words, the two-term control described by Garrard is found using the lowest-order nonzero f_n term, whatever that may be.

This method has some limitations. It is not possible to increase the accuracy of the approximation by adding more terms to the power series without the tedious process of solving for the polynomial of the appropriate order. Only the first two terms of the series can be found by matrix calculations as described above. Also, the two-term control considers only one order of nonlinearity, as shown in equation (4.6) where only f_2 is used, ignoring any higher order parts of f. If a problem has both quadratic and cubic nonlinearities, for example, then the cubic part of the system will be ignored. Due to this property, it seems reasonable to only use this method on problems with one order of nonlinearity (whether this order is quadratic, cubic, or higher). The method does have the advantage that it is very easy to implement, and its calculations can be done very quickly.

4.3.2 State-Dependent Riccati Equation

This method also results in a feedback control in power series form, though it does so through a different process. Instead of solving the HJB equation itself, the power series expansion is applied to the related state-dependent Riccati equation (SDRE). The SDRE method is discussed in detail by Cloutier, D'Souza and Mracek in [71], and the use of the power series for this equation is from Wernli and Cook [69].

The idea behind the method is to parallel the use of the Riccati equation for linear problems by rewriting the nonlinear function of x in the system (4.1) as $f(x) = A(x)x$. Note that the choice of A is not unique, and different choices will result in different controls. With f rewritten in this way the state-dependent Riccati equation is of the form

$$\Pi(x)A(x) + A^T(x)\Pi(x) - \Pi(x)BR^{-1}B^T\Pi(x) + Q = 0, \tag{4.7}$$

and the optimal feedback control is given by

$$u(x) = -R^{-1}B^T\Pi(x)x. \tag{4.8}$$

We note that the Riccati solution $\Pi(x)$ is state-dependent and is not as easy to find as for the constant-valued case, except for simple problems with certain structures. One proposed method for the general case is to use a power series expansion to approximate the solution.

We rewrite A in terms of a constant part and a state-dependent part as $A(x) = A_0 + \varepsilon\Delta A(x)$, where ε is a temporary variable used for the expansion that will be set to 1 later. We next write Π as a power series in ε,

$$\begin{aligned}\Pi(x,\varepsilon) &= \Pi(x)|_{\varepsilon=0} + \Pi_\varepsilon(x)|_{\varepsilon=0}\varepsilon + \Pi_{\varepsilon\varepsilon}(x)|_{\varepsilon=0}\frac{\varepsilon^2}{2} + ... \\ &= \sum_{n=0}^{\infty} \varepsilon^n L_n(x), \end{aligned} \tag{4.9}$$

where Π is symmetric, and so each L_n is symmetric as well. Substituting these expansions into the state-dependent Riccati equation (4.7), we find

$$\begin{aligned}\left(\sum_{n=0}^{\infty}\varepsilon^n L_n\right)\left[A_0 + \varepsilon\Delta A(x)\right] + \left[A_0^T + \varepsilon\Delta A^T(x)\right]\left(\sum_{n=0}^{\infty}\varepsilon^n L_n\right) \\ - \left(\sum_{n=0}^{\infty}\varepsilon^n L_n\right) BR^{-1}B^T\left(\sum_{n=0}^{\infty}\varepsilon^n L_n\right) + Q &= 0.\end{aligned}$$

Now by matching terms involving the same powers of ε we obtain the following set of equations that can be used to determine L_n:

$$L_0A_0 + A_0^TL_0 - L_0BR^{-1}B^TL_0 + Q = 0, \tag{4.10}$$

$$L_1\left(A_0 - BR^{-1}B^TL_0\right) + \left(A_0^T - L_0BR^{-1}B^T\right)L_1 + L_0\Delta A + \Delta A^TL_0 = 0, \tag{4.11}$$

$$\begin{aligned}L_n\left(A_0 - BR^{-1}B^TL_0\right) + \left(A_0^T - L_0BR^{-1}B^T\right)L_n + L_{n-1}\Delta A + \Delta A^TL_{n-1} \\ - \sum_{k=1}^{n-1}\left(L_kBR^{-1}B^TL_{n-k}\right) &= 0.\end{aligned} \tag{4.12}$$

Equation (4.10) is the standard Riccati equation for the linear part A_0. Equations (4.11) and (4.12) may be solved, but as with the HJB power series expansion it may be a tedious process if the function $\Delta A(x)$ is complicated. However, in some cases it is easier to obtain a higher-order control for the SDRE expansion than for the HJB expansion. For example, in the case that the nonlinear function $f(x)$ in (4.1) is quadratic in x, the function $\Delta A(x)$ will be linear in x, and from (4.11) the L_1 term will be of order x which will produce an order x^2 control, and so on. In comparison, the HJB expansion requires the order x^3 term V_1 to obtain an order x^2 control; thus it is necessary to consider a much larger number of possible combinations of the state variables in x.

The SDRE method is much easier to consider for a certain class of nonlinearity factorizations, specifically, those for which ΔA has the same function of x in all of its elements, and so can be written as $\Delta A(x) = g(x)\Delta A_C$ with a constant-valued matrix ΔA_C. In this case, by defining $L_n(x) = g^n(x)(L_n)_C$, where $(L_n)_C$ is a constant matrix, we obtain from (4.12) the equation

$$\begin{aligned}(L_n)_C\left(A_0 - BR^{-1}B^TL_0\right) + \left(A_0^T - L_0BR^{-1}B^T\right)(L_n)_C + (L_{n-1})_C\Delta A_C + \Delta A_C^T(L_{n-1})_C \\ - \sum_{k=1}^{n-1}\left[(L_k)_CBR^{-1}B^T(L_{n-k})_C\right] &= 0.\end{aligned}$$

This is just a constant-valued matrix Lyapunov equation, for which solvers are readily available, as with the constant-valued Riccati equation. In this way a large class of SDRE problems can be solved to as many terms of the power series as desired. Once as many L_n terms as desired have been found, the control is obtained by substituting them back into equations (4.9) and (4.8) with ε set to 1.

The SDRE method is similar in its advantages and disadvantages to the two-term HJB method. Both consist of calculating out the first few terms of a power series, and both can be calculated very quickly. However, as many terms of the SDRE series as desired can be found for the case when $\Delta A = g(x)\Delta A_C$, as described above, while the HJB approach always uses only two terms. This assumption on the form of ΔA does limit the problems for which the SDRE approach is most useful, just as the HJB approach is most useful for problems with only one level of nonlinearity. There is also the drawback that the SDRE method is approximating the solution of the SDRE, which would produce only a suboptimal control even if solved exactly, while the HJB method is approximating the HJB equation itself.

4.3.3 Successive Galerkin Approximation

Instead of the regular HJB equation (4.2), the successive Galerkin approximation (SGA) method (from Beard, Saridis and Wen in [65, 66]) uses the generalized Hamilton-Jacobi-Bellman (GHJB) equation. Because there is an error in the paper by Beard et al [66] on page 602, we restate the proposed method here in detail. Consider the cost functional

$$V(x_0; u) = J(x_0, u) = \int_0^\infty (x^T Q x + u^T R u) dt$$

where $x(t)$ satisfies the nonlinear dynamics $\dot{x} = f(x) + Bu(x)$, with control $u(x)$ and initial condition x_0. Differentiating V along the path of the system leads to the GHJB equation,

$$\text{GHJB}[V, u] \equiv \left(\frac{\partial V}{\partial x_0}(x_0)\right)^T [f(x_0) + Bu(x_0)] + x_0^T Q x_0 + u^T(x_0) R u(x_0) = 0. \tag{4.13}$$

This becomes the normal HJB equation if the formula for the control,

$$u(x_0) = -\frac{1}{2} R^{-1} B^T \frac{\partial V}{\partial x_0}(x_0), \tag{4.14}$$

is used in (4.13). Instead, a process is set up to solve equations (4.13) and (4.14) together in an iterative manner.

We first choose an initial control function $u^{(0)}(x)$ (for example, the optimal linear feedback control). Then we use the GHJB equation (with the variable x_0 now changed to x) to find $\partial V^{(0)}/\partial x$ by solving

$$\left(\frac{\partial V^{(0)}}{\partial x}(x)\right)^T \left[f(x) + Bu^{(0)}(x)\right] + x^T Q x + \left[u^{(0)}(x)\right]^T R u^{(0)}(x) = 0.$$

We let the next control in the sequence be

$$u^{(1)}(x) = -\frac{1}{2} R^{-1} B^T \frac{\partial V^{(0)}}{\partial x}(x).$$

We proceed to find $V^{(1)}$ and $u^{(2)}$ in the same way, and continue like this for as long as desired.

A Galerkin method can be used to approximate the GHJB equation so that it can be solved numerically. To do this, we choose a set of N basis functions $\{\phi_j\}_{j=1}^N$ and define the $\mathbf{L}_2$ inner product as

$$\langle f, g \rangle = \int_\Omega f(x) g(x) dx,$$

where Ω is any compact subset of the region of attraction associated with the known stabilizing control $u^{(0)}(x)$. The Galerkin approximation of $V^{(i)}$ is given by

$$V_N^{(i)}(x) = \sum_{j=1}^N c_j^{(i)} \phi_j(x),$$

where the coefficients $c_j^{(i)}$ satisfy the weak form of (4.13):

$$\left\langle \text{GHJB}\left[\sum_{j=1}^N c_j^{(i)} \phi_j, u_N^{(i)}\right], \phi_n \right\rangle = 0 \tag{4.15}$$

for $n = 1, ..., N$. The initial control function $u_N^{(0)}$ for (4.15) is chosen, and later iterates are found as follows. Equation (4.15) gives N equations to be solved for the unknowns $c_j^{(i)}$, $j = 1, ..., N$:

$$\sum_{j=1}^N c_j^{(i)} \left\langle \left(\frac{\partial \phi_j}{\partial x}\right)^T \left(f + B u_N^{(i)}\right), \phi_n \right\rangle + \left\langle x^T Q x + (u_N^{(i)})^T R u_N^{(i)}, \phi_n \right\rangle = 0 \tag{4.16}$$

for $n = 1, ..., N$. The next control update becomes

$$u_N^{(i+1)} = -\frac{1}{2} R^{-1} B^T \sum_{j=1}^N c_j^{(i)} \left(\frac{\partial \phi_j}{\partial x}\right). \tag{4.17}$$

From equations (4.16) and (4.17) the process can be written as an iterative matrix problem, with the necessary inner products calculated beforehand. We will use the notation $C_N^{(i)} = \left[c_1^{(i)}, ..., c_N^{(i)}\right]^T$

and $\Phi_N(x) = [\phi_1(x), ..., \phi_N(x)]^T$, and let $\nabla\Phi_N$ denote the Jacobian of Φ_N. Also, for any real-valued function $\eta(x)$, define a vector of inner products

$$\langle \eta, \Phi_N \rangle_v = [\langle \eta, \phi_1 \rangle, ..., \langle \eta, \phi_N \rangle]^T,$$

and for any vector-valued function $\eta(x) = (\eta_1(x), ..., \eta_N(x))^T$, define a matrix of inner products

$$\langle \eta, \Phi_N \rangle_m = \begin{bmatrix} \langle \eta_1, \phi_1 \rangle & \cdots & \langle \eta_N, \phi_1 \rangle \\ \vdots & \ddots & \vdots \\ \langle \eta_1, \phi_N \rangle & \cdots & \langle \eta_N, \phi_N \rangle \end{bmatrix}.$$

Compute the following L_2 inner products:

$$I^Q = \left\langle x^T Q x, \Phi_N \right\rangle_v, \; I^R = \left\langle (u^{(0)})^T R u^{(0)}, \Phi_N \right\rangle_v, \; I^f = \langle \nabla\Phi_N f, \Phi_N \rangle_m, \tag{4.18}$$

$$I^B = \left\langle \nabla\Phi_N B u^{(0)}, \Phi_N \right\rangle_m, \; \left\{ I_k^{BRB} \right\}_{k=1}^N = \left\{ \left\langle \nabla\Phi_N B R^{-1} B^T (\partial\phi_k/\partial x), \Phi_N \right\rangle_m \right\}_{k=1}^N. \tag{4.19}$$

From the chosen initial iterate $u^{(0)}$, solve

$$A c_N^{(0)} = b,$$

where $A = I^f + I^B$ and $b = -I^Q - I^R$. Given the ith iterate, the $(i+1)$th iterate is the solution to the equation

$$A c_N^{(i+1)} = b,$$

where $A = I^f - (1/2)\sum_{k=1}^N c_k^{(i)} I_k^{BRB}$ and $b = -I^Q - (1/4)\left(\sum_{k=1}^N c_k^{(i)} I_k^{BRB}\right) C_N^{(i)}$. (Note the 1/4 in the expression for b, which was incorrectly written as 1/2 in the original paper [66].) At each step $i \geq 0$, the control is given by

$$u_N^{(i+1)} = -\frac{1}{2} R^{-1} B^T \nabla\Phi_N^T C_N^{(i)}.$$

The successive Galerkin approximation method is more time-intensive than the first two methods discussed. The actual iteration process is fast, but the time needed to calculate the integrals in (4.18) and (4.19) can be significant. The method has the advantage that it is applicable to a larger class of problems, without the restrictions imposed by the first two methods. However, being an iterative method, it has the disadvantage that it is dependent on the initial iterate $u^{(0)}$. If this is not well chosen, then the method may converge very slowly, or may not even converge at all.

4.3.4 Interpolation of TPBV Problem Solutions

A very different approach to finding a feedback control is used in the remaining two methods to be discussed. These methods use techniques for finding an open-loop control for the system with a chosen initial condition x_0. This is repeated for many values of x_0. At each point, the initial control $u^{(x_0)}(t=0)$ that is found is considered the control value at x_0. Then an interpolation is done over all these initial points to form a closed-loop (feedback) control $u(x)$. The open-loop control is found by solving the Hamiltonian system arising from the necessary optimality conditions [57, 75] given by

$$\begin{cases} \dot{x} &= f(x) - \frac{1}{2}BR^{-1}B^T p \\ \dot{p} &= -f_x^T(x)p - 2Qx, \end{cases} \tag{4.20}$$

with initial condition $x(0) = x_0$ and final condition $\lim_{t\to\infty} p(t) = 0$. From this, the open-loop control is calculated as

$$u(t) = -\frac{1}{2}R^{-1}B^T p(t).$$

In the version of this method described by Ito and Schroeter [74], this problem is solved as follows. First the final time is changed from infinity to a chosen finite value T; then the two-point boundary value (TPBV) problem is discretized numerically for $t \in [0, T]$ with a mixed finite difference scheme. The discrete TPBV problem is given by

$$\begin{cases} \frac{1}{\Delta t}(x_k - x_{k-1}) &= \frac{1}{2}\left[f(x_k) + f(x_{k-1})\right] - \frac{1}{2}BR^{-1}B^T p_k \\ \frac{1}{\Delta t}(p_{k+1} - p_k) &= -\frac{1}{2}f_x^T(x_k)\,(p_k + p_{k+1}) - 2Qx_k, \end{cases} \tag{4.21}$$

where $k = 1, ..., N-1$, and N is a prescribed number of discretization intervals, so that $\Delta t = T/N$. Here x_0 is the chosen initial condition and $p_N = 0$.

Letting $y = (x_1, ..., x_{N-1}, p_1, .., p_{N-1})$ and rewriting the system (4.21) as

$$F(y) = \begin{pmatrix} \frac{1}{\Delta t}(x_k - x_{k-1}) - \frac{1}{2}\left[f(x_k) + f(x_{k-1})\right] + \frac{1}{2}BR^{-1}B^T p_k \\ \frac{1}{\Delta t}(p_{k+1} - p_k) + \frac{1}{2}f_x^T(x_k)\,(p_k + p_{k+1}) + 2Qx_k \end{pmatrix}_{k=1,...,N-1} = 0 \tag{4.22}$$

results in a $2m(N-1)$ dimensional system of nonlinear equations (where m is the number of state variables) which can be solved using Newton's method. The Newton iterate is

$$y_{n+1} = y_n - \left[\nabla F(y_n)\right]^{-1} F(y_n),$$

where ∇F is the Jacobian of F and has the following structure:

$$\nabla F = \begin{bmatrix} A & C \\ D & -A^T \end{bmatrix}.$$

The matrix A is block lower bi-diagonal and the matrices C and D are block diagonal, all with block size m. The values of the blocks are as follows (using I_m as an $m \times m$ identity matrix):

$$\begin{aligned} A_{k,k} &= \frac{1}{\Delta t} I_m - \frac{1}{2} f_x(x_k), \\ A_{k+1,k} &= -\frac{1}{\Delta t} I_m - \frac{1}{2} f_x(x_k), \\ C_{k,k} &= \frac{1}{2} B R^{-1} B^T, \\ D_{k,k} &= \frac{1}{2} f_{xx}^T(x_k)\,(p_k + p_{k+1}) + 2Q. \end{aligned}$$

We have carried out the required interpolations, following Ito and Schroeter's choice, with Green's functions of the type $G(x,z) = |x-z|^\alpha$ and a choice of $\alpha = 3.7$ for the exponent. The closed-loop control is then given by

$$u(x) = \sum_{i=1}^{M} \eta_i G\left(x, (x_0)_i\right),$$

where $\{\eta_i\}$ are the coefficients of the interpolation and $\{(x_0)_i\}$ are the M initial points used to interpolate over the region Ω. The coefficients are chosen so that the control fits exactly at the interpolation points, so that

$$u((x_0)_j) = \sum_{i=1}^{M} \eta_i G\left((x_0)_j, (x_0)_i\right) = u_{OL}(t=0, (x_0)_j)$$

for all $j = 1, ..., M$. Here $u_{OL}(t=0,(x_0)_j)$ is the open-loop control for the initial condition $(x_0)_j$, evaluated at $t = 0$. The coefficients $\{\eta_i\}$ that will achieve this exact fit at the interpolation points are found by solving the $2M$ dimensional linear system

$$\begin{bmatrix} I_K G((x_0)_1,(x_0)_1) & \cdots & I_K G((x_0)_1,(x_0)_M) \\ \vdots & \ddots & \vdots \\ I_K G((x_0)_M,(x_0)_1) & \cdots & I_K G((x_0)_M,(x_0)_M) \end{bmatrix} \begin{bmatrix} \eta_1 \\ \vdots \\ \eta_M \end{bmatrix} = \begin{bmatrix} u_{OL}(t=0,(x_0)_1) \\ \vdots \\ u_{OL}(t=0,(x_0)_M) \end{bmatrix},$$

where I_K is an identity matrix of dimension K, the number of control variables. Note that the values on the diagonal are all zero for the chosen function G.

The use of the interpolation makes this method very time-consuming, given that open-loop controls must be found for many different initial states. This is especially true for problems with a larger number of state variables; this increases the time needed by making the discretized TPBV problem much larger and by increasing the number of interpolation points needed due to the larger dimension. It is not a very complicated process, however, and all of the expensive computation is done offline, so the actual control of the system can still be done very quickly. There are no

restrictions on the types of problem that can be solved. The process of increasing the accuracy of the control by adding more interpolation points is also straightforward.

4.3.5 Interpolation of Iterative Solutions

As in the previous method, this technique uses interpolation over a number of open-loop control solutions, but the method of obtaining these solutions is different. In this method, proposed by Markman and Katz [78, 79], the Hamiltonian system is solved not as one large discretized TPBV problem given by (4.22), but through an iteration process leading to the desired control.

We again choose a final time T, and with it we now define a distance function

$$F(T, p_0) = \|x(T; x_0, p_0)\|^2 + \|p(T; x_0, p_0)\|^2 = \|z(T; z_0)\|^2 .$$

Here $z = (x, p)^T$ is used to simplify notation, and $z(T; z_0)$ is the solution at time T of the Hamiltonian system (4.20) starting at z_0; $x(T; x_0, p_0)$ and $p(T; x_0, p_0)$ are the state and costate parts of this solution. F is a measurement of how close to zero the system will be at time T, given a value of p_0. Thus the open-loop control is found by solving $\inf_{p_0} F(T, p_0)$ to find the value of p_0 which minimizes F (and so drives the system closest to zero at time T). The minimization is done by a Gauss-Newton method. We start with a chosen $p_0^{(0)}$ and iterate as follows:

$$\left[p_0^{(j+1)}\right]^T = \left[p_0^{(j)}\right]^T - \left[\nabla F(p_0^{(j)})\right]^T \left[\nabla^2 F(p_0^{(j)})\right]^{-1}, \tag{4.23}$$

where ∇F and $\nabla^2 F$ are the gradient and Hessian of F with respect to p_0 and are given by

$$\begin{aligned}
\nabla F(T, p_0^{(j)}) &= \left[\frac{\partial z}{\partial p_0}(T, p_0^{(j)})\right]^T z(T, p_0^{(j)}), \\
\nabla^2 F(T, p_0^{(j)}) &= \left[\frac{\partial z}{\partial p_0}(T, p_0^{(j)})\right]^T \left[\frac{\partial z}{\partial p_0}(T, p_0^{(j)})\right] + z^T(T, p_0^{(j)}) \left[\frac{\partial^2 z}{\partial p_0^2}(T, p_0^{(j)})\right].
\end{aligned}$$

For a sufficiently close initial iterate, the term $\partial^2 z/\partial p_0^2$ is assumed to be very small, so it is ignored. This means that equation (4.23) becomes

$$\left[p_0^{(j+1)}\right]^T = \left[p_0^{(j)}\right]^T - z^T \frac{\partial z}{\partial p_0} \left[\frac{\partial z^T}{\partial p_0}\frac{\partial z}{\partial p_0}\right]^{-1}.$$

This iteration process requires both z and $\partial z/\partial p_0$ at the final time T. The value of z can be found simply by solving the Hamiltonian system (4.20) forward to T, starting from the initial condition $z_0 = (x_0, p_0)^T$. The partial derivatives can be found in the following way. If the Hamiltonian

system is $\dot{z} = \overline{H}z + h(z)$, with $z_0 = (x_0, p_0)^T$, then differentiating with respect to p_0 produces

$$\begin{cases} \frac{d}{dt}\frac{\partial z}{\partial p_0} &= \left[\overline{H} + \frac{\partial h}{\partial z}(z)\right]\frac{\partial z}{\partial p_0} \\ \frac{\partial z}{\partial p_0}(0) &= \begin{bmatrix} 0 \\ I \end{bmatrix}. \end{cases}$$

Adjoining this to the Hamiltonian system results in a $2m(m+1)$-dimensional system of ODEs to be solved on $[0, T]$.

In the method of Markman and Katz, the minimization of $F(T, p_0)$ over p_0 is actually carried out for a sequence of time values T increasing toward infinity. This results in a sequence of values for p_0 which converge on the final value as the finite-time problems get closer to the actual infinite-time problem. The solution of the Hamiltonian system generated by the final p_0 in the sequence is then used to obtain the desired control $u(t)$. This process is important because the values of p_0 early in the sequence may not be very good, and may cause the system to blow up if T is chosen too large. Starting with a small T and increasing it from there helps to avoid this. It is also useful in showing that the values of p_0 which are found actually are converging as $T \to \infty$. The TPBV method described in Section 4.3.4 does not need to be solved for a sequence of final times, because there is no threat of the problem blowing up. However, it could be useful (though time-consuming) to do this to make certain that T has not been chosen too small to give a good approximation of the infinite-time problem.

The initial value of p_0 can be chosen from the linear quadratic regulator problem. That is, $p_0^{(0)} = 2\Pi x_0$, where Π is the solution to the algebraic Riccati equation. For subsequent values of T in the sequence, the previous final iterate can be used as the new initial iterate. The interpolation of the open-loop controls can be carried out in the same way as described in Section 4.3.4, using the previously specified Green's functions with $\alpha = 3.7$.

As with the previous method, the interpolation makes the calculation of this feedback control a very time-consuming process. However, the iterative method of finding the open-loop control should be somewhat faster than the discretized TPBV problem, especially as the problem dimension increases. It can also be refined by adding more interpolation points in the same way as the previous method. There is no strict limit on the types of problem to which it can be applied, but as with the SGA method of Section 4.3.3, the performance of this method depends in an important way on the choice of the initial iterate.

4.4 Application to Test Problems

In this section we will compare the performance of the five methods discussed in the previous section on several test examples. We will consider their control authority (i.e., ability to drive the states to zero), the amount of control used, the overall cost of the methods, and the computational time and flops (floating point operations) used. All computations were done using MATLAB codes (written by the author, using the built-in functions "*are*" for solving constant-valued algebraic Riccati equations and "*ode45*" for solving ODE systems), implemented on a Sun Ultra 5.

The SDRE method was implemented using 5 power series terms in the following simulations. In the Ito and Schroeter TPBV problem method, the iteration process was run until $\|F\| < 10^{-5}$. The Markman and Katz iterative method was cut off when $\|z_i(T) - z_{i-1}(T)\| < 10^{-5}$, for each T, and $\|z(T_j) - z(T_{j-1})\| < 10^{-5}$ in the sequence of T values. The particular sequence of T values used varies among the test problems, as do the discretization interval Δt and final time T used in the TPBV problem method. Another parameter that varies is the region Ω over which the SGA method integrates and the TPBV and iterative interpolation (to be called Iter in discussion of the following examples) methods interpolate, with the number of interpolation points varying as well.

4.4.1 Example 1: Simple Problem

The first example is described by Cloutier, D'Souza and Mracek in [90], and contains two state variables and two control variables. The system, which has cubic nonlinearities in each equation, is given by

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} x_1 - x_1^3 + x_2 \\ x_1 + x_1^2 x_2 - x_2 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}.$$

We factor the nonlinear function $f(x) = A(x)x$ by

$$f(x) = \begin{bmatrix} 1 - x_1^2 & 1 \\ 1 & x_1^2 - 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

for the SDRE method. For the problem, we used the initial condition $x_0 = [1, 1]^T$. The cost functional to be minimized is

$$J(x_0, u) = \int_0^\infty \left(x^T \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{bmatrix} x + u^T \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} u \right) dt.$$

For the TPBV method, our calculations used the time step $\Delta t = 0.05$ and the final time $T = 5.25$. For the iterative method, the sequence of T values was started at $T_0 = 1.0$ and increased by equally

spaced time intervals of $\Delta T = 0.25$ up to the final $T = 5.25$. The region $\Omega = [-1, 1] \times [-1, 1]$ and $100 = 10 \times 10$ regularly spaced interpolation points were used.

Figure 4.1 shows the state norm $\|x\|$ for the five feedback controls described earlier, as well as the open-loop control found using the discretized two point boundary value problem approach for the initial condition x_0. All of these results are extremely close to each other, with the TPBV

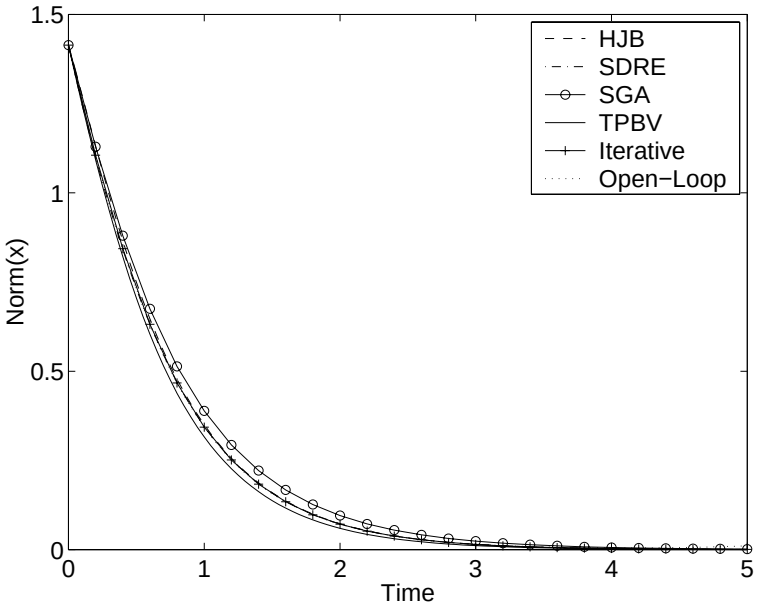

Figure 4.1: Comparison of the norms of feedback controlled trajectories in Example 1.

method performing slightly better, and the SGA method performing slightly worse. The norms of the control $\|u\|$ are shown in Figure 4.2 for all five methods. There are only small differences between most of the methods, with the clear exception being the SGA method, which requires nearly twice as much control.

The time and flops required for each feedback control, along with each of their values for the cost functional $J(x_0, u)$, are listed in Table 4.1. To compare, the open-loop control cost is 4.7965, and the linear control (which also succeeds in driving the problem to zero in this example) cost is 4.7810. The costs are all very similar (except SGA), varying by only about 0.5% among themselves, at about 5% less than the linear control. The computational time for the TPBV and Iter methods, involving the solution of many open-loop problems, is much larger than that of the HJB and SDRE methods. The computational time for the SGA method is between these extremes, but is still much less than

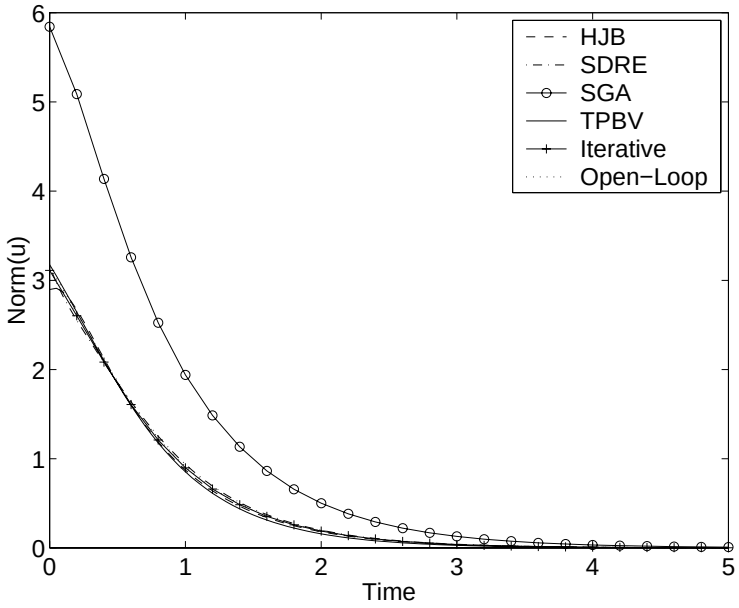

Figure 4.2: Comparison of the norms of feedback controls in Example 1.

for the interpolation methods. However, the expensive computations of SGA, TPBV and Iter are all done offline, so they could be run again very quickly with a different x_0 if so desired. For this problem, the best choice of methodology appears to be HJB or SDRE: they achieve results virtually as good as any other method, and do so with minimal computation.

4.4.2 Example 2: Simple Problem

The second example chosen to test the various feedback control methods is from Markman and Katz [78]. It has two state variables, one control variable, and a cubic nonlinearity in the system,

	CPU Time (sec.)	Flops	Cost
HJB	0.15	4.71×10^4	4.6985
SDRE	0.21	4.89×10^4	4.6929
SGA	1.52	1.69×10^6	17.5309
TPBV	448.46	1.62×10^{10}	4.6809
Iter	974.89	2.28×10^8	4.6768

Table 4.1: Numerical comparison of feedback control methodologies in Example 1.

which is defined by

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ x_1^3 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u.$$

The function $f(x)$ is rewritten as

$$f(x) = \begin{bmatrix} 0 & 1 \\ x_1^2 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

for the SDRE method. The initial condition is $x_0 = [1, 0]^T$ and the cost functional to be minimized is

$$J(x_0, u) = \int_0^\infty \left(x^T \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{bmatrix} x + \frac{1}{2} u^2 \right) dt.$$

For this example, in the TPBV method we used $\Delta t = 0.05$ and $T = 7.50$. For the iterative interpolation method we used $T_0 = 1.0$ and $\Delta T = 0.5$. The same region $\Omega = [-1, 1] \times [-1, 1]$ as in Example 1 was used, with $100 = 10 \times 10$ interpolation points.

Figure 4.3 depicts the state x_1 and Figure 4.4 the control u for all five control methods. The SGA method iterations never actually converged to a final control; the one shown here (which

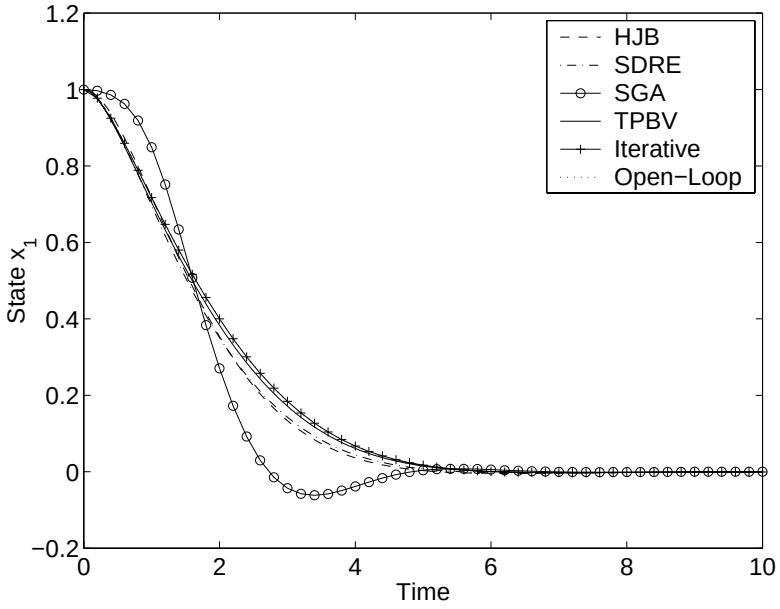

Figure 4.3: Comparison of feedback controlled states x_1 in Example 2.

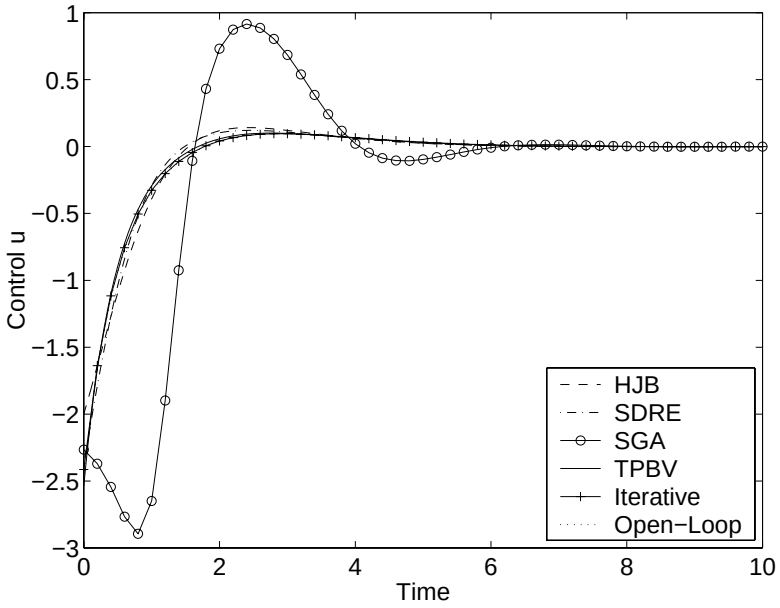

Figure 4.4: Comparison of feedback controls in Example 2.

does force the system to zero) is taken after an iteration number truncation. It can be seen in the figures that the interpolation methods drive the states to zero slightly more slowly than the SDRE and HJB methods, but that the SDRE and HJB methods require slightly more control to achieve this.

Table 4.2 contains the time required, flops required, and cost for each control method. To

	CPU Time (sec.)	Flops	Cost
HJB	0.25	8.43×10^4	1.5028
SDRE	0.30	9.26×10^4	1.5629
SGA	2.19	2.10×10^6	5.5935
TPBV	1413.3	4.90×10^{10}	1.4867
Iter	1065.9	2.47×10^8	1.4842

Table 4.2: Numerical comparison of feedback control methodologies in Example 2.

compare, the open-loop control cost is 1.5624. For this example the linear control fails to bring the system to zero. The costs are similar (again with the exception of SGA), but not as close as for Example 1. They vary by about 5% among themselves and are effective when the linear

control fails. The computational times are very much like those in Example 1, with the times for TPBV and Iter again much larger than those for HJB and SDRE, and with SGA in between. The best choice of control here might be HJB: its performance is almost as good as the interpolation methods, without the intensive computations, and its cost is less than that of SDRE.

With this example we also used different initial conditions to see how the various controls would perform with starting points further away from the origin. We tested both $x_0 = [2, 0]^T$ and $x_0 = [4, 0]^T$. Note that both of these points are outside the region Ω used for the interpolations in the TPBV and Iter methods and the integrations in the SGA method. Thus, some loss of effectiveness in these methods should be expected. The HJB and SDRE methods should also suffer, since their power series approximations will be less accurate further away from the expansion point of the origin.

It turns out that, for both initial conditions, the SDRE and SGA methods fail to stabilize the system. The value of the cost functional J for the other three methods, and the open-loop control calculated through the TPBV problem, are displayed in Table 4.3 for both initial conditions. The

	Cost for $[2, 0]^T$	Cost for $[4, 0]^T$
HJB	28.19	1340
TPBV	25.73	2170
Iter	26.95	5720
Open-Loop	27.66	982

Table 4.3: Feedback control methodologies in Example 2 with a distant initial state.

state x_1 and control u for $x_0 = [4, 0]^T$ are graphed in Figures 4.5 and 4.6. The results for $x_0 = [2, 0]^T$ are not too different from those for $x_0 = [1, 0]^T$, at least for those methods that are still effective: the cost differences between the three feedback controls and the open-loop control expand somewhat, up to nearly a 10% difference between them, and the open-loop control is relatively more effective than before. However, for $x_0 = [4, 0]^T$ there are dramatic differences. The open-loop control is much more effective, which is to be expected, since it is being solved for this initial point specifically rather than depending on an interpolation region or power series expansion point which is far away. The HJB method is also shown to be much better than TPBV (which is in turn much better than Iter), in terms of stabilization, amount of control, and cost. For this example at least, the HJB method produces the feedback control least affected by the choice of an initial state away from the origin.

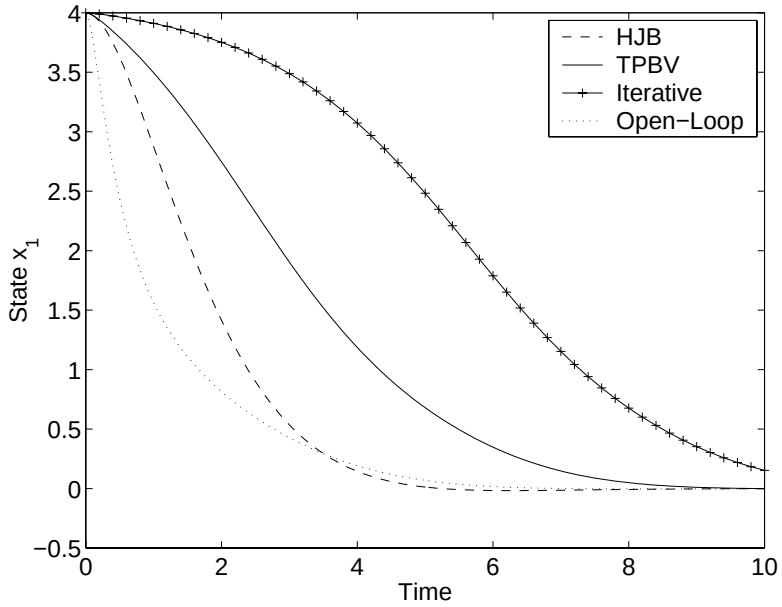

Figure 4.5: Feedback controlled states x_1 in Example 2 with a distant initial state.

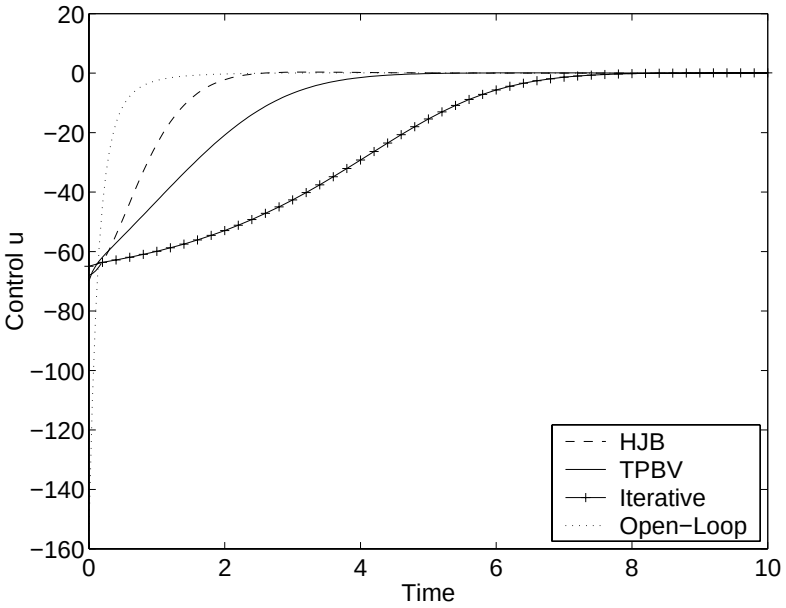

Figure 4.6: Feedback controls in Example 2 with a distant initial state.

Since the two-term HJB method performed very well for this example, we also calculated more terms in the power series to see how much is lost by restricting it to two terms. This is complicated, as mentioned in Section 4.3.1, but less so for this example since the system is relatively simple. Fifth-order and seventh-order terms were added to the first-order and third-order terms already found, and the resulting control was used on the example with initial condition $x_0 = [1, 0]^T$. The result was slightly better than the two-term control in its effect on the state variables, but used slightly more control and resulted in a cost of 1.5060 compared to the cost for the two-term control of 1.5028. For this example, we can conclude that very little is lost by the restriction to two terms.

4.4.3 Example 3: Larger Flight Dynamics Model

The third example that we studied is a model of the flight dynamics of a high-performance aircraft taken from Garrard, Enns and Snell [61]. Of the test problems in this chapter, this is the most like the HPCVD problem: it is a higher-dimensional problem than the first two, containing five state variables and one control variable, and it has quadratic nonlinearities only. We have changed the problem from the description in [61] by scaling the variables so that they are closer to the same order. This was done by converting from degrees to radians in x_5 and u in order to match the units of the other variables, and by changing x_1 from units of m/s to 100m/s. We also changed the cost functional to one of the form that we are considering (quadratic in the state and control).

The state variables in this model represent the flight conditions of the aircraft: x_1 is the deviation of the velocity from the level flight trim value of 1(100m/s), x_2 is the deviation of the angle of attack (the angle of the aircraft body with respect to the velocity direction) from the trim value of $4.2(\pi/180)$ radians, x_3 is the pitch rate (the rate of change in the angle of the aircraft body) in rad/s, x_4 is the flight path angle (the angle of the velocity direction with respect to horizontal) in radians, and x_5 is the deviation of the canard deflection angle in radians from the trim value, which is not given. The control u is the input canard deflection in radians. The canards are control flaps which can deflect downward by up to $90(\pi/180)$ radians; there is a lag in the input given to them included in the model. The initial condition chosen for this example is a high angle of attack x_2 with otherwise trim values; the control methods will attempt to stabilize the aircraft by driving all the variables to the trim conditions. (The meaning of the model is described in more detail in [61] for interested readers.)

The system is given by

$$\dot{x} = (A_0 + x_2 A_{NL})\, x + Bu,$$

where the matrices A_0, A_{NL} and B are all constant-valued and are given by:

$$A_0 = \begin{bmatrix} -0.0443 & 1.1280 & 0.0 & -0.0981 & 0.0 \\ -0.0490 & -2.5390 & 1.0 & 0.0 & -0.0854 \\ -0.0730 & 19.3200 & -2.2700 & 0.0 & 22.6834 \\ 0.0490 & 2.5390 & 0.0 & 0.0 & 0.0854 \\ 0.0 & 0.0 & 0.0 & 0.0 & 20.0 \end{bmatrix},$$

$$A_{NL} = \begin{bmatrix} -0.2317 & 0.0 & 0.0 & 0.0 & 0.0 \\ -1.2760 & -0.7922 & 0.0 & 0.0 & 0.0206 \\ 0.1020 & 64.2940 & -13.9710 & 0.0 & -5.4167 \\ 1.2760 & 0.7922 & 0.0 & 0.0 & -0.0206 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \end{bmatrix},$$

$$B = \begin{bmatrix} 0.0 & 0.0 & 0.0 & 0.0 & 20.0 \end{bmatrix}^T.$$

The initial condition is $x_0 = [0, 25(\pi/180), 0, 0, 0]^T$. The cost functional to be minimized is

$$J(x_0, u) = \int_0^\infty \left(x^T I_5 x + 100u^2\right) dt,$$

where the matrix I_5 is a 5×5 identity matrix. For this example, we used $\Delta t = 0.15$, $T = 15.00$, $T_0 = 0.5$ and $\Delta T = 0.25$. The region used here is $\Omega = [-0.3, 0.3] \times [-25(\pi/180), 25(\pi/180)]^4$, with $1024 = 4^5$ interpolation points.

This example appears to be much more difficult to solve than the previous ones. This may be caused by the larger dimension of the problem, or because the system dynamics are more highly unstable than in the previous examples, or because the chosen initial state is far enough away from zero to make the problem very difficult. The SGA method does not converge. The Iter algorithm for finding the open-loop control for a given initial point fails to converge for many of the points in the interpolation domain. The TPBV method does find the open-loop controls for all the interpolation points, but the feedback control created by the interpolation fails to drive the system to zero.

Figure 4.7 depicts the state x_2, which is the primary variable that we want to control and the one which is initially nonzero. Figure 4.8 displays the norm $\|x\|$ of the entire state vector to provide a better view of the overall effect of the control methods on the system. Figure 4.9 depicts the control u. These graphs are all done for HJB, SDRE, the open-loop control found by the TPBV problem discretization, and the linear control, all of which successfully stabilized the system. The HJB method performs best on both the state x_2 and the norm $\|x\|$ overall. SDRE performs somewhat

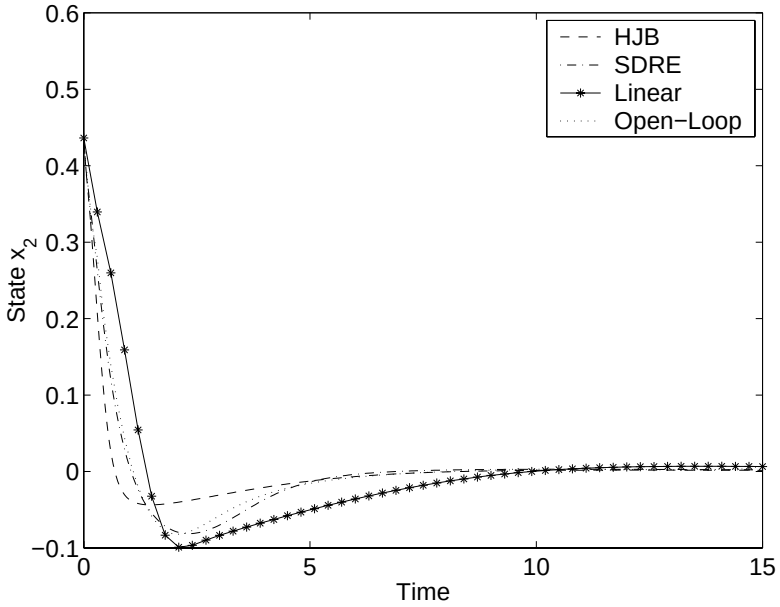

Figure 4.7: Comparison of feedback controlled states x_2 in Example 3.

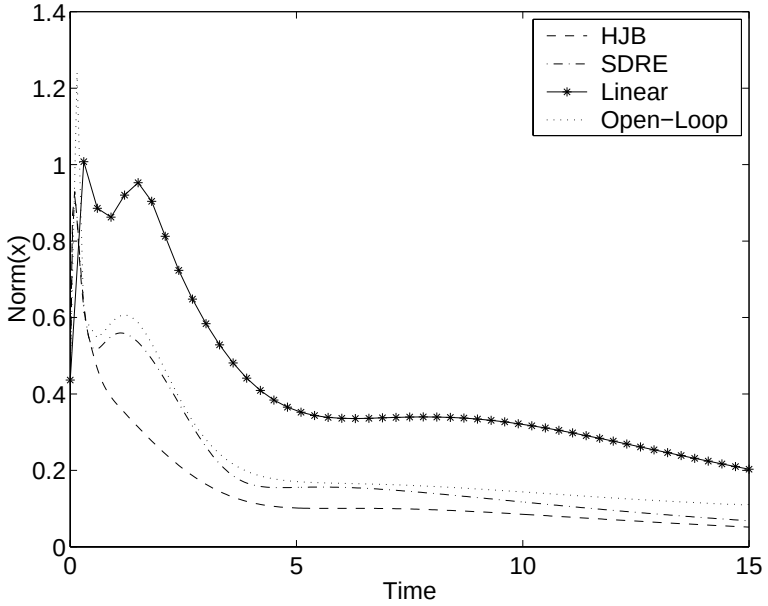

Figure 4.8: Comparison of the norms of feedback controlled trajectories in Example 3.

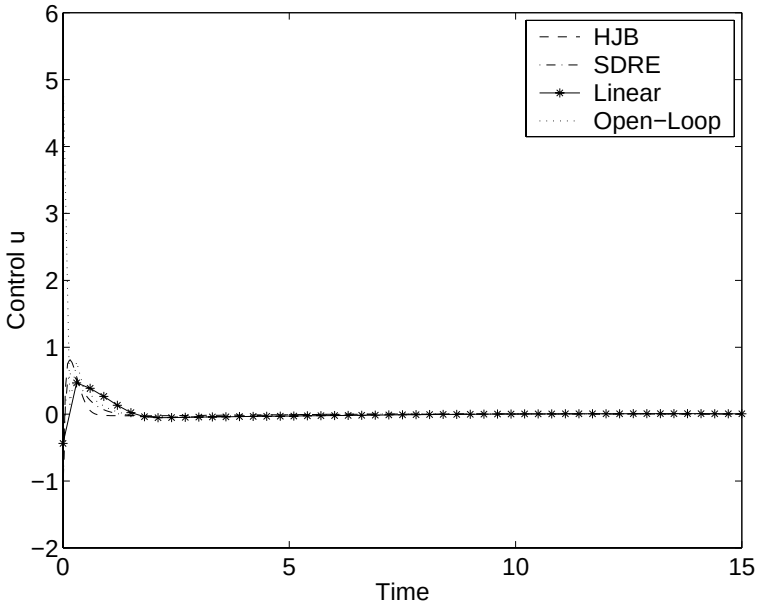

Figure 4.9: Comparison of feedback controls in Example 3.

better than the open-loop control with $\|x\|$ but slightly worse with x_2. All three methods are more effective than the linear control in terms of stabilization. In comparison, the control used by the HJB method is larger than that of SDRE for the initial burst, but then lower after a time of about 0.5. Similarly, SDRE uses more control than the linear control in the initial burst, but less thereafter. The open-loop control is very large initially before dropping down to the level of the other controls. This may explain why the interpolation of these open-loop TPBV controls failed to produce an effective closed-loop control. The interpolation is carried out on the initial value of the control, which in this case is very large in comparison with the feedback controls and even in comparison with the open-loop control later in time. This may make it difficult to extend with an interpolation, especially when it is done so roughly, with only four points in each dimension.

The time and flops required and the value of the cost functional J for the two successful nonlinear feedback controls are listed in Table 4.4. The open-loop control cost for this problem is 107.58, very large compared with the others due to the large initial control which is weighted heavily in the cost functional. Interestingly, the linear control for this example has a cost of only 10.2013, apparently because it has very little of the heavily-weighted control compared with the other methods. However, the stabilization effect of the linear control is smaller compared with the other methods, as shown

	CPU Time (sec.)	Flops	Cost
HJB	0.88	1.44×10^6	13.7983
SDRE	1.10	1.09×10^6	11.3175

Table 4.4: Numerical comparison of feedback control methodologies in Example 3.

in Figures 4.7 and 4.8.

Determining which method is best is more difficult here than for the first two examples. The SDRE and linear control have lower costs but are less effective in stabilizing the state variables, while the HJB control appears to be better in terms of state stabilization but results in a higher cost. The larger weight on the control in the cost functional causes this higher cost by amplifying the small differences in the controls seen in Figure 4.9 into large cost differences. The choice of the large weight in the cost functional was made to avoid nonphysical results such as the huge initial spike seen in the open-loop control. This spike was successfully avoided for the other methods, but the large weight also results in the HJB method having a higher cost even though it uses only slightly more control. So, even though the linear control has the lowest cost, the HJB method could well be considered the best choice for this example, based on its effect on the states. This is unlike in Example 1, where the cost and state behavior match up well, and in Example 2, where the effect of the control on the cost functional was based on a greater difference between the methods.

4.4.4 Example 4: Flight Model with Quadratic and Cubic Nonlinearities

This example is taken from Garrard and Jordan [60] and is a different model of flight control. It has only three state variables instead of five, but contains both quadratic and cubic nonlinear terms. In this model the state variables are as follows: x_1 is the deviation of the angle of attack from the trim value of 0.044 radians, x_2 is the flight path angle in radians, and x_3 is the rate of change of the flight path angle in rad/s. The control u is the deviation of the tail deflection angle (with respect to the aircraft body direction) from the trim value of -0.009 radians. In the aircraft described by this model, the entire tail can rotate to control its flight. As in the previous example, the control methods are asked to stabilize the system from an initial condition of a high angle of attack x_1 with otherwise trim conditions. (The interested reader can find more information on this model in [60].)

The system is written as

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} -0.877x_1 + x_3 + 0.47x_1^2 - 0.088x_1x_3 - 0.019x_2^2 + 3.846x_1^3 - x_1^2x_3 \\ x_3 \\ -4.208x_1 - 0.396x_3 - 0.47x_1^2 - 3.564x_1^3 \end{bmatrix} + \begin{bmatrix} -0.215 \\ 0.0 \\ -20.967 \end{bmatrix} u$$

with an initial condition $x_0 = [25(\pi/180), 0, 0]^T$. For the SDRE method, the nonlinear function is rewritten as

$$f(x) = \begin{bmatrix} -0.877 + 0.47x_1 + 3.846x_1^2 & -0.019x_2 & 1 - 0.088x_1 - x_1^2 \\ 0 & 0 & 1 \\ -4.208 - 0.47x_1 - 3.564x_1^2 & 0 & -0.396 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}.$$

The cost functional is

$$J(x_0, u) = \int_0^\infty \left(x^T \frac{1}{4} I_3 x + u^2 \right) dt.$$

For this example, we used $\Delta t = 0.1$ and $T = 12.0$ in the TPBV method, and $T_0 = 0.5$ and $\Delta T = 0.25$ in the iterative method. The region $\Omega = [-25(\pi/180), 25(\pi/180)]^3$ was used, and there were $125 = 5^3$ interpolation points.

As with the other flight problem in Example 3, neither of the iterative methods converged. The TPBV method, which failed for that problem, does work here. Figure 4.10 depicts the state x_1 and Figure 4.11 displays the control u for HJB, SDRE, TPBV, and the open-loop control found by the TPBV problem discretization. The TPBV method is clearly the best at controlling x_1, with the open-loop control a close second; both of these have a large initial control which quickly becomes very small. In contrast, the SDRE and HJB methods use more constant control, with less control than the other methods at first but more after a time of 0.5 or so. They are also significantly worse at controlling x_1, with the HJB method the worst of the four. This should be expected, since this is our first example with a relatively complicated nonlinear structure. The two-term HJB method ignores the cubic terms in the system, as explained in Section 4.3.1, using only the linear and quadratic parts. The nonlinear part of the system cannot be rewritten for the SDRE method as $\Delta A = g(x)\Delta A_C$, as described in Section 4.3.2, so the power series is difficult to compute for that method. Only two terms were used in this example, compared with five terms for the other problems. The TPBV method does not have such strong drawbacks with this type of problem, so it is naturally much more effective than the other feedback controls.

The time, flops, and value of the cost functional J for each control method are displayed in Table 4.5. The open-loop control has a cost of 0.0483, and the linear control has a cost of 0.0576. These match the results seen in Figures 4.10 and 4.11, with the TPBV method slightly better than

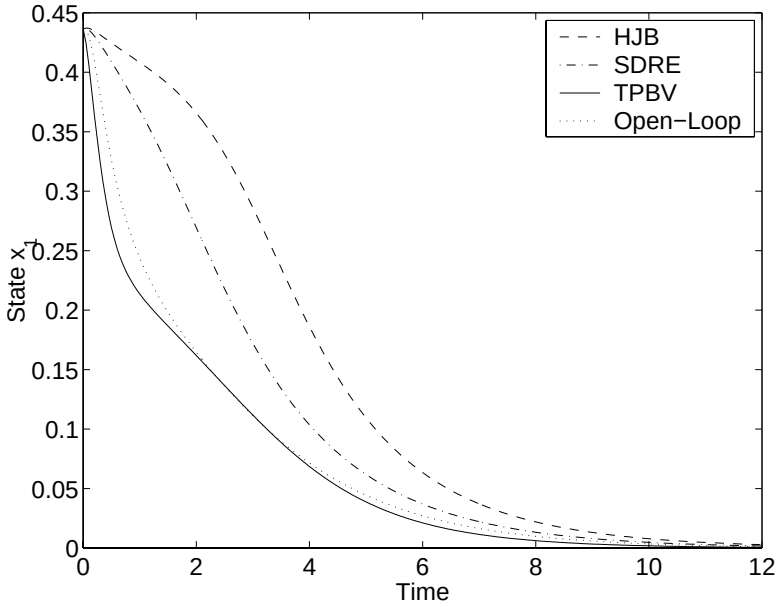

Figure 4.10: Comparison of feedback controlled states x_1 in Example 4.

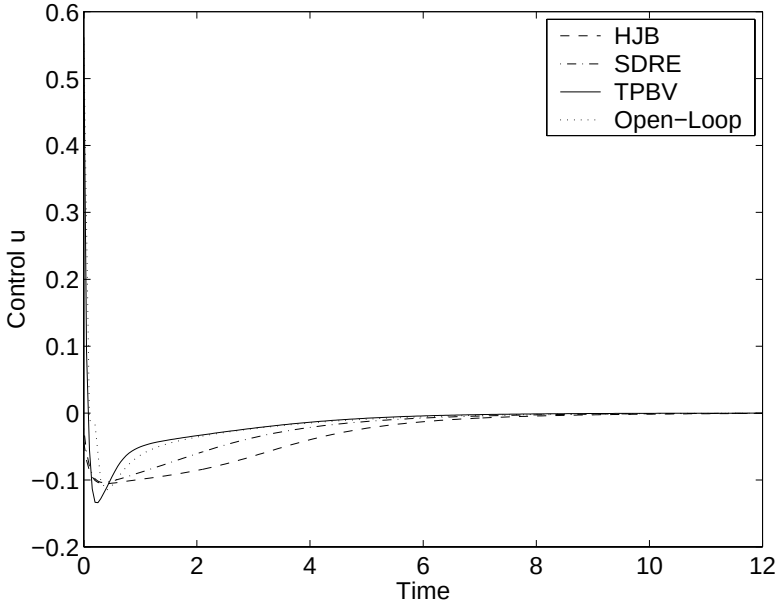

Figure 4.11: Comparison of feedback controls in Example 4.

	CPU Time (sec.)	Flops	Cost
HJB	0.42	2.29×10^5	0.0856
SDRE	0.41	1.85×10^5	0.0578
TPBV	6014	2.14×10^{11}	0.0435

Table 4.5: Numerical comparison of feedback control methodologies in Example 4.

the open-loop, with SDRE significantly less effective, and HJB clearly last in performance. The TPBV interpolation method takes a very long time to do all the open-loop calculations it needs, but for this example it is well worth the effort, since it produces much better results than the other feedback controls.

4.5 Conclusions

In comparing several feedback control methods for nonlinear systems, we have seen that for many problems it appears that a simple method is the best choice. The method of Garrard using only two power series terms is very effective for problems with one level of nonlinearity, a category including many problems of great interest. For Examples 1-3, which are all of this type, the two-term HJB method produced results nearly as good or better than the other feedback control methods examined. It is very easy to implement and does not require the much larger computational time which some of the other methods need. It also proved more effective in Example 2 for initial states further away from the origin. For problems with more complex nonlinearities, such as Example 4, the two-term HJB method is not very effective, so a less restricted method is a better choice. In this case the TPBV interpolation method produced good results, although requiring a great deal of computational time. The more complicated iterative methods had difficulties with some of the examples, failing to converge perhaps because of inadequate initial iterates. Convergence results are given in [66, 79] for the iterative methods, but these require restrictions, for example on the initial iterate, which do not necessarily hold for our examples.

Based on these results, the HJB method seems to be the one we would choose to apply to the high-pressure chemical vapor deposition problem. The HPCVD problem has only quadratic nonlinearities, which the HJB method can handle very well. The large size of the problem would result in a very large amount of computational time needed for the preparations for the interpolation methods (and to a lesser extent the SGA method), but the HJB method is very quick and easy even

for large problems. However, as we will see next chapter, the HJB control is not well suited to being extended to include tracking control and state estimation, which are necessary parts of the HPCVD problem. Instead we use the SDRE method, which shares many of the characteristics of the HJB method: it is fast, easy to implement, and effective at controlling the test problems most like the HPCVD problem.

Chapter 5

State Estimation and Tracking Control of Nonlinear Dynamical Systems

5.1 Introduction

The previous chapter consisted of a comparative survey of several different methods of designing a feedback control for nonlinear systems. Since finding the exact optimal feedback control is, in general, impossible for the nonlinear case, the methods in the last chapter used various approaches towards finding an effective suboptimal control. In this chapter we will extend one of the nonlinear feedback control methods to treat problems involving tracking control and state estimation, which would allow it to be used in a wide class of real applications. Our emphasis will be on the development of computational methods for constructing nonlinear estimators and nonlinear feedback tracking controls that are easily implementable as well as efficient at controlling problems similar to the HPCVD model.

Both tracking and estimation, like the simple optimal feedback control problem, have been widely studied for linear systems, and methods of formulation of the control and estimator are well known (for example, see [57, 58] and the references they contain). In the linear case the tracking problem is solved in two parts: a state feedback control determined by the algebraic Riccati equation (ARE), and a time-dependent tracking variable found by solving an ordinary differential equation incorporating the desired trajectory backwards from a stable final time. The state estimation problem for linear systems involves the formulation of a state estimator system using the observed measurements of the actual state and a gain matrix found through a second Riccati equation.

To construct observers and tracking controls for the nonlinear case we at first hoped to extend

the two-term power series control of Garrard [59], discussed in Section 4.3.1, given its simplicity and effectiveness in the types of problems studied in Section 4.4. However, since it is based directly on the Hamilton-Jacobi-Bellman equation for optimal feedback control, and the linear tracking control in particular is derived from the Hamiltonian state/costate formulation of the control problem, combining the two proved difficult. Instead, our approaches are based on the state-dependent Riccati equation (SDRE) [67, 68, 69, 70, 71], discussed in Section 4.3.2, which is also very simple to calculate, though overall not quite as effective as the two-term power series method at controlling the chosen test problems in Section 4.4. However, as shown in the following sections, the SDRE method is more readily adaptable to the nonlinear tracking and nonlinear state estimation problems, since it is closely related to the ARE-based methods used to find those controls in the linear problem.

While there is a large amount of literature available on state estimation for nonlinear systems, a literature search revealed very little material on tracking control for nonlinear systems. The tracking control technique discussed in this chapter was developed directly out of the combination of the SDRE for nonlinear control with the linear method for tracking control. The method of state estimation we will describe below is closely related to work by Thau and others [91, 92, 93, 94] on nonlinear estimators which are themselves extensions of the linear state estimation formula.

There are other very different methods for state estimation described in the literature which we decided not to use, mostly because they are difficult to implement, though theoretically solid. A large effort has been made with types of methods that use a nonlinear transformation to change the subject system into a very specific observer form, mostly using many Lie derivatives of the nonlinear functions from the problem measurement and dynamics [95, 96, 97, 98, 99, 100, 101]. Other methods use Lie derivatives as well but not in a transformation to a simpler system form [102, 103]. Also, there are methods using a linearization about a family of constant operating points of the system [104], variable-structure system techniques [105], and Lyapunov theory [106]. A number of these methods are compared in the survey paper of Walcott, Corless and Zak [107]. These methods prove difficult to use in many applications, especially problems with complicated nonlinearities in the dynamics or measurement. More specifically, the nonlinear transformation is often very hard to find or may not even exist, higher-order Lie derivatives are often hard to calculate, the linearization around the family of solutions may be very difficult, and finding the appropriate Lyapunov function may be quite challenging. There may be restrictions on the types of problems for which various methods can guarantee results theoretically, or even be used at all. The state-dependent Riccati equation based method that we develop involves some approximations and assumptions, but it is

straightforward to implement and is applicable to a class of nonlinear systems (including the reduced order HPCVD system) that are important in practice.

In Section 5.2 we will look more closely at the derivation of the state-dependent Riccati equation (in the context of simple stabilizing feedback control of nonlinear systems) than when it was previously discussed in Section 4.3.2. Section 5.3 will focus on our formulation of the feedback tracking control, and Section 5.4 on the state estimator, for the nonlinear case. The nonlinear estimator and tracking control will be implemented on two test problems in Section 5.5. Their performance will be compared with that of the linear estimator and tracking control found through the linearized system. Some overall conclusions about these methods will be given in Section 5.6.

5.2 The State-Dependent Riccati Equation

In this section we will give a little more detail on the state-dependent Riccati equation (SDRE) than in Section 4.3.2. We will specifically look at how it is derived, to provide context for the similar derivation of the nonlinear tracking control in Section 5.3. Early work on the SDRE was done by Pearson [67] and Burghart [68], and it is well described in detail by Cloutier, D'Souza and Mracek in [71]. The SDRE is simply an extension of the constant-valued algebraic Riccati equation used to find the optimal feedback control in the linear quadratic regulator problem.

Consider a system of the type

$$\begin{cases} \dot{x}(t) &= f(x(t)) + Bu(x(t)) \\ x(0) &= x_0, \end{cases} \tag{5.1}$$

which is nonlinear in the state $x \in \Omega \subset R^m$ and is linear in the control $u : \Omega \to R^k$, with a quadratic cost functional

$$J(x_0, u) = \frac{1}{2} \int_0^\infty \left(x^T Q x + u^T R u \right) dt$$

with given constant-valued weighting matrices Q and R of appropriate dimensions. The optimal control problem is to find a state feedback control $u^*(x^*)$ which minimizes the cost for all possible initial conditions x_0. (We should point out the factor of $1/2$ included in the cost functional here. In Chapter 4 we did not use this, in order to remain consistent with most of the papers from which we took the nonlinear control methods. However, the notation in [71] as well as [58] does have the $1/2$, so we include it in this chapter to match them.)

The derivation of the SDRE begins with the Hamiltonian for this problem, which is defined as

$$\mathcal{H}(x,u,p) = \frac{1}{2}x^T Q x + \frac{1}{2}u^T R u + p^T\left(f(x) + Bu\right).$$

From the Hamiltonian the necessary conditions for the optimal control are given by

$$\dot{x}^* = \frac{\partial \mathcal{H}}{\partial p} = f(x^*) + Bu^*, \tag{5.2}$$

$$\dot{p}^* = -\frac{\partial \mathcal{H}}{\partial x} = -Qx^* - \frac{\partial f^T}{\partial x}(x^*)p^*, \tag{5.3}$$

$$0 = \frac{\partial \mathcal{H}}{\partial u} = Ru^* + B^T p^*. \tag{5.4}$$

From equation (5.4), the control is given in terms of the costate, p^*, by $u^*(t) = -R^{-1}B^T p^*(t)$.

We will seek a costate of the form $p^*(t) = \Pi\left(x^*(t)\right)x^*(t)$. First, taking the derivative of p^* yields $\dot{p}^* = \Pi\left(x^*\right)\dot{x}^* + D_t\Pi\left(x^*\right)x^*$. Then we substitute into this equation the formulas for $\dot{x}^*$ and $\dot{p}^*$ determined by the necessary conditions (5.2) and (5.3), and the formulas for u^* and p^* given above, resulting in

$$\Pi\left(x^*\right)\left[f(x^*) - BR^{-1}B^T\Pi\left(x^*\right)x^*\right] + D_t\Pi\left(x^*\right)x^* = -Qx^* - \frac{\partial f^T}{\partial x}(x^*)\Pi\left(x^*\right)x^*,$$

which can be rewritten as

$$\Pi\left(x^*\right)f(x^*) + \frac{\partial f^T}{\partial x}(x^*)\Pi\left(x^*\right)x^* - \Pi\left(x^*\right)BR^{-1}B^T\Pi\left(x^*\right)x^* + Qx^* + D_t\Pi\left(x^*\right)x^* = 0. \tag{5.5}$$

The term $D_t\Pi(x)$ is a somewhat misleading notation; it is the total time derivative of $\Pi(x(t))$ given by

$$D_t\Pi(x) = \sum_{k=1}^{m}\frac{\partial \Pi}{\partial x_k}(x)\dot{x}_k,$$

and thus has no meaning except when evaluated along a state trajectory $x(t)$ so that $\dot{x}$ can have some value.

For the simpler linear problem, where the dynamics are $f(x) = A_0 x$, x^* can be factored out of each term in equation (5.5) so that it becomes

$$\Pi A_0 + A_0^T\Pi - \Pi BR^{-1}B^T\Pi + Q = 0. \tag{5.6}$$

Equation (5.6) is now the standard algebraic Riccati equation with a constant-valued solution matrix Π. The resulting optimal feedback control is given by $u^*(x^*) = -R^{-1}B^T\Pi x^*$.

The SDRE method involves mimicking the above use of the Riccati equation by rewriting the nonlinear function of x in (5.1) as $f(x) = A(x)x$ (which is not a unique choice, as mentioned in

Section 4.3.2). With f rewritten in this way, equation (5.5) becomes

$$\begin{aligned}\Pi(x^*)\,A(x^*)x^* + A^T(x^*)\Pi(x^*)\,x^* + \sum_{i=1}^{m} x_i^* \left(\frac{\partial A_{1\to m,i}}{\partial x}(x^*)\right)^T \Pi(x^*)\,x^* \\ - \Pi(x^*)\,BR^{-1}B^T\Pi(x^*)\,x^* + Qx^* + D_t\Pi(x^*)\,x^* &= 0,\end{aligned}$$

which can then be rewritten as

$$\begin{aligned}\Big[\Pi(x^*)\,A(x^*) + A^T(x^*)\Pi(x^*) - \Pi(x^*)\,BR^{-1}B^T\Pi(x^*) + Q \\ + \sum_{i=1}^{m} x_i^* \left(\frac{\partial A_{1\to m,i}}{\partial x}(x^*)\right)^T \Pi(x^*) + D_t\Pi(x^*)\Big]x^* &= 0,\end{aligned}$$

where the A-column derivatives are given by

$$\frac{\partial A_{1\to m,i}}{\partial x} = \begin{bmatrix} \partial A_{1i}/\partial x_1 & \cdots & \partial A_{1i}/\partial x_m \\ \vdots & \ddots & \vdots \\ \partial A_{mi}/\partial x_1 & \cdots & \partial A_{mi}/\partial x_m \end{bmatrix}. \tag{5.7}$$

This equation can be divided into the state-dependent Riccati equation

$$\Pi(x)A(x) + A^T(x)\Pi(x) - \Pi(x)BR^{-1}B^T\Pi(x) + Q = 0 \tag{5.8}$$

and two extra terms

$$\sum_{i=1}^{m} x_i \left(\frac{\partial A_{1\to m,i}}{\partial x}(x)\right)^T \Pi(x) + D_t\Pi(x) = 0. \tag{5.9}$$

In the SDRE method one assumes the extra terms remain small and ignores them (thus, among other things, removing the necessity of specifying a trajectory $x(t)$ so that $D_t\Pi(x)$ can be evaluated in the equation). This assumption creates a suboptimal feedback control based on the solution to the Riccati equation (5.8) and given by

$$u(x) = -R^{-1}B^T\Pi(x)x. \tag{5.10}$$

To find the SDRE solution $\Pi(x)$ of (5.8) we follow the power series expansion method of Wernli and Cook [69], as described in Section 4.3.2. The technique of splitting A into constant and state-dependent parts by $A(x) = A_0 + \varepsilon g(x)\Delta A_C$ is applicable to both of the test problems we use in Section 5.5, as well as the HPCVD system described in Sections 2 and 3. The power series solution to the SDRE is then substituted into (5.10) to obtain the nonlinear feedback control law.

Some theoretical results are available for the state-dependent Riccati equation and the suboptimal feedback control law derived from it. Wernli and Cook [69] show the existence and asymptotic stability of the controlled system, given certain assumptions on the properties of the system. Cloutier,

D'Souza and Mracek [71] prove, given certain assumptions, properties of local and global stability, robustness and suboptimality for the control. Hammett, Hall and Ridgely [108] discuss the issues of controllability and stabilizability (properties important to the proofs in [71]) in regards to the SDRE method of controlling nonlinear systems.

The SDRE method can be calculated very quickly for the case with $\Delta A(x) = g(x)\Delta A_C$, but this assumption on the form of ΔA does limit the problems for which the SDRE approach is most useful. There is also the drawback that this method of control ignores the extra terms in equation (5.9) in setting up the SDRE, and additionally the power series solution is only an approximation to the exact solution of the SDRE (in particular it is less accurate further from the power series expansion point of the origin). These approximations lead to an only suboptimal feedback control, although the application of the method to test problems in Section 4.4 shows generally very good results.

5.3 Tracking Control for Nonlinear Systems

In this section we will derive an SDRE-based solution to the feedback tracking control problem for a nonlinear system in a manner analogous to the derivation in [58] for a linear system. We focus on the differential equation system

$$\begin{cases} \dot{x}(t) &= f(x(t)) + Bu(x(t),t) \\ x(0) &= x_0 \\ y(t) &= Hx(t). \end{cases}$$

This is the same as the problem studied in the previous section with the addition of the variable y, which is the signal we want to follow along a desired trajectory. We take y as a linear function of the state variables x for now. The cost function for the tracking problem, with a desired trajectory $r(t)$, is given by

$$J(x_0, u) = \frac{1}{2}\int_0^\infty \left[(y-r)^T Q(y-r) + u^T R u\right] dt.$$

To prepare for the application of the SDRE, we rewrite the nonlinear function as $f(x) = A(x)x$. With this notation, the Hamiltonian is given by

$$\mathcal{H}(x,u,p) = \frac{1}{2}(Hx-r)^T Q(Hx-r) + \frac{1}{2}u^T R u + p^T\left(A(x)x + Bu\right).$$

The necessary conditions for the optimal control are

$$\dot{x}^* = \frac{\partial \mathcal{H}}{\partial p} = A(x^*)x^* + Bu^*, \tag{5.11}$$

$$\dot{p}^* = -\frac{\partial \mathcal{H}}{\partial x} = -H^T Q(Hx^* - r) - A^T(x^*)p^* - \sum_{i=1}^{m} x_i^* \left(\frac{\partial A_{1\to m,i}}{\partial x}(x^*) \right)^T p^*, \tag{5.12}$$

$$0 = \frac{\partial \mathcal{H}}{\partial u} = Ru^* + B^T p^*, \tag{5.13}$$

with the A-column derivatives as defined in equation (5.7). From equation (5.13), the control is given by the form $u^*(t) = -R^{-1}B^T p^*(t)$.

We seek a costate of the form $p^*(t) = \Pi\left(x^*(t)\right) x^*(t) + s(t)$ (with a time-dependent tracking variable s added to the version of the costate in Section 5.2) which satisfies the necessary conditions. As before, we take the derivative of p^* to obtain $\dot{p}^* = \Pi\left(x^*\right)\dot{x}^* + \dot{s} + D_t\Pi\left(x^*\right)x^*$. Then we substitute in the formulas for $\dot{x}^*$ and $\dot{p}^*$ from the necessary conditions (5.11) and (5.12), and the above formulas for u^* and p^*, yielding

$$\begin{aligned}
&\Pi\left(x^*\right)\left[A(x^*)x^* - BR^{-1}B^T\left(\Pi\left(x^*\right)x^* + s\right)\right] + \dot{s} + D_t\Pi\left(x^*\right)x^* = \\
&\quad - H^T Q(Hx^* - r) - A^T(x^*)\left(\Pi\left(x^*\right)x^* + s\right) - \sum_{i=1}^{m} x_i^* \left(\frac{\partial A_{1\to m,i}}{\partial x}(x^*) \right)^T \left(\Pi\left(x^*\right)x^* + s\right),
\end{aligned}$$

which can be rewritten as

$$\begin{aligned}
&\left[\Pi\left(x^*\right)A(x^*) + A^T(x^*)\Pi\left(x^*\right) - \Pi\left(x^*\right)BR^{-1}B^T\Pi\left(x^*\right) + H^T QH\right]x^* + \Big[\dot{s} + A^T(x^*)s \\
&- \Pi\left(x^*\right)BR^{-1}B^T s - H^T Qr + \sum_{i=1}^{m} x_i^* \left(\frac{\partial A_{1\to m,i}}{\partial x}(x^*) \right)^T \left(\Pi\left(x^*\right)x^* + s\right) + D_t\Pi\left(x^*\right)x^*\Big] \quad = \quad 0.
\end{aligned}$$

This separates the equation into two parts: a state-dependent Riccati equation for determining $\Pi(x)$ with which to find the feedback control gain, and an ODE for determining the tracking variable $s(t)$.

In the non-tracking feedback control problem in Section 5.2, the terms involving derivatives of A and Π in equation (5.9) were assumed to be small and were neglected, thus reducing the problem down to Riccati equation form. This neglecting of terms is what makes the SDRE method only a suboptimal means of control. However, now there is a second part to the problem, the tracking variable equation given by

$$\dot{s} + A^T(x)s - \Pi\left(x\right)BR^{-1}B^T s - H^T Qr + \sum_{i=1}^{m} x_i \left(\frac{\partial A_{1\to m,i}}{\partial x}(x) \right)^T \left(\Pi\left(x\right)x + s\right) + D_t\Pi\left(x\right)x = 0. \tag{5.14}$$

We note that the solution to this equation is state-dependent (i.e., s should actually be written as $s(t,x)$) through the presence of $A(x)$ and $\Pi(x)$ even without the derivative terms, so by keeping the

derivative terms in this equation we can include their effects in the control design without drastically changing the nature of the problem.

Solving the tracking equation is more difficult for this nonlinear problem than for the linear case described in [58]. In the linear problem the ODE is solved, offline, backwards from an assumed final time value $s(T_f) = 0$. For the nonlinear case, $s(t, x)$ must still be computed offline (since the value of s at $t = 0$ is unknown), but now this computation must include the dependence of the equation on x through the state-dependent A and Π as well as the derivative terms not present in the linear case. We do this by solving for s using a state trajectory $x_{nom}(t)$ found with the state equation $\dot{x}_{nom} = A(x_{nom})x_{nom} - BR^{-1}B^T(\Pi\,(x_{nom})\,x_{nom} + s)$ coupled to equation (5.14). Given the initial condition $x_{nom}(0) = x_0$ and the final condition $s(T_f, x_{nom}(T_f)) = 0$, this results in a two-point boundary value problem which can be discretized and solved with a mixed finite difference method (described below) for $x_{nom}(t)$ and $s(t, x_{nom}(t))$. Note that the coupling of the s and x_{nom} equations provides the trajectory $x_{nom}(t)$ necessary to evaluate the $D_t\Pi(x_{nom})$ term in equation (5.14).

We use the notation x_{nom} to clarify that this is a nominal trajectory found offline, before actual implementation of the feedback control on the system. If the system dynamics are known precisely then it will match the actual trajectory precisely. However, for example, in Section 5.5 the control is applied to a problem with random noise in the $\dot{x}$ system but not in the nominal $\dot{x}_{nom}$ system, to represent a problem where the actual physical system $\dot{x}$ has some unpredictable noise or other dynamics which the model $\dot{x}_{nom}$ is unable to include. In the offline calculation of s coupled with x_{nom} in the TPBV problem, any differences between x and x_{nom} due to an altered initial value or some variation or noise will result in an inaccurate s since it is found using an inaccurate x_{nom}. We tacitly assume that x_{nom} is a good prediction of the actual state behavior, and that any small differences between it and x will not drastically impact the effectiveness of the control. Another area of concern in this tracking control method is the fact, discussed in Section 5.2, that since we are finding a power series based solution to the SDRE, it will be more inaccurate the further the state is from the expansion point, which is the origin. This is particularly important for the tracking problem, since we specifically want the state to follow a certain nonzero trajectory.

The numerical discretization of the two-point boundary value problem for s and x_{nom} on the interval $t \in [0, T_f]$ is done in a manner very similar to that used on the state/costate TPBV problem in Section 4.3.4. The x variables (changed from x_{nom} in the following description for clarity) have an initial condition, so a backward difference formula is applied to them, while the s variables have a final condition, so a forward difference formula is applied to them. This leads to the discrete

system

$$\begin{cases} \frac{1}{\Delta t}\left(s_{k+1}-s_k\right) &= -\frac{1}{2}\left[A^T(x_k)-\Pi(x_k)BR^{-1}B^T\right]\left(s_k+s_{k+1}\right)+\frac{1}{2}H^TQ\left(r_k+r_{k+1}\right) \\ & \quad -\sum_{i=1}^{m}(x_k)_i\left(\frac{\partial A_{1\to m,i}}{\partial x}(x_k)\right)^T\left[\Pi\left(x_k\right)x_k+\frac{1}{2}\left(s_k+s_{k+1}\right)\right] \\ & \quad -\frac{1}{\Delta t}\left[\Pi(x_{k+1})-\Pi(x_k)\right]x_k \\ \frac{1}{\Delta t}\left(x_k-x_{k-1}\right) &= \frac{1}{4}\left[A(x_k)+A(x_{k-1})-BR^{-1}B^T\left(\Pi(x_k)+\Pi(x_{k-1})\right)\right]\left(x_k+x_{k+1}\right) \\ & \quad -BR^{-1}B^Ts_k, \end{cases} \tag{5.15}$$

where $s_k = s(k\Delta t)$ and $x_k = x_{nom}(k\Delta t)$, for discretization points $k = 1, ..., N-1$. N is the chosen number of discretization intervals, so that $\Delta t = T_f/N$. Here x_0 is the given initial condition and $s_N = 0$ is the final condition. The x variables are not averaged in the s equation, and the s variables are not averaged in the x equation, because we have no values for x_N or s_0. However, the discretization of $D_t\Pi\left(x\right)$ needs two time values to find the difference version of the derivative, and thus includes an x_N value. For this we replace $\left[\Pi(x_N)-\Pi(x_{N-1})\right]/\Delta t$ by $\left[\Pi(x_{N-1})-\Pi(x_{N-2})\right]/\Delta t$, making the assumption that this change will be small, since near T_f the state will be stable and there will be very little change in x there. The discretized system can be written in terms of the variables $y = [s_1, .., s_{N-1}, x_1, ..., x_{N-1}]$ as $F(y) = 0$, resulting in a $2m(N-1)$-dimensional system of nonlinear equations. This can be solved using a Newton's method formula, with the next iterate found from the current one by

$$y_{n+1} = y_n - \left[\nabla F(y_n)\right]^{-1}F(y_n),$$

where ∇F is the Jacobian of F.

To summarize, the control formula

$$u(x,t) = -R^{-1}B^T\left[\Pi\left(x\right)x + s(t, x_{nom})\right] \tag{5.16}$$

is found by, first, solving the SDRE

$$\Pi\left(x\right)A(x) + A^T(x)\Pi\left(x\right) - \Pi\left(x\right)BR^{-1}B^T\Pi\left(x\right) + H^TQH = 0 \tag{5.17}$$

for $\Pi\left(x\right)$ in the manner described in Section 4.3.2. The second part of the control design process is then to find the tracking variable $s(t, x_{nom})$ from

$$\begin{cases} \dot{s} &= -A^T(x_{nom})s + \Pi\left(x_{nom}\right)BR^{-1}B^Ts + H^TQr - D_t\Pi\left(x_{nom}\right)x_{nom} \\ & \quad -\sum_{i=1}^{m}\left(x_{nom}\right)_i\left(\frac{\partial A_{1\to m,i}}{\partial x_{nom}}(x_{nom})\right)^T\left(\Pi\left(x_{nom}\right)x_{nom}+s\right) \\ \dot{x}_{nom} &= A(x_{nom})x_{nom} - BR^{-1}B^T(\Pi\left(x_{nom}\right)x_{nom}+s) \end{cases} \tag{5.18}$$

in a two-point boundary value problem with $x_{nom}(0) = x_0$ and $s(T_f, x_{nom}(T_f)) = 0$, as discretized in the system (5.15).

For a nonlinear tracking signal $y(t) = H(x(t))x(t)$, the control problem becomes somewhat more complicated. The state-dependence of H affects the control formulation in a similar way to that of A, resulting in the Riccati and tracking variable equations in (5.17) and (5.14) being expanded into the following forms:

$$\begin{aligned} \Pi(x)\,A(x) + A^T(x)\Pi(x) - \Pi(x)\,BR^{-1}B^T\Pi(x) + H^T(x)QH(x) &= 0, \qquad (5.19) \\ \dot{s} + A^T(x)s - \Pi(x)\,BR^{-1}B^Ts + \sum_{i=1}^{m} x_i \left(\frac{\partial H_{1\to n,i}}{\partial x}(x)\right)^T Q\left(H(x)\,x - r\right) & \\ - H^T(x)Qr + \sum_{i=1}^{m} x_i \left(\frac{\partial A_{1\to m,i}}{\partial x}(x)\right)^T \left(\Pi(x)\,x + s\right) + D_t\Pi(x)\,x &= 0. \end{aligned}$$

The tracking variable equation is only slightly changed, adding one more term and making H state-dependent in another; neither of these changes strongly affect the solution method. The SDRE contains the added state-dependence of H in the H^TQH term, which can have more important consequences. If $H(x)$ is not a very complicated formula as was discussed for $A(x)$ in Section 4.3.2, such as $H(x) = H_0 + g(x)\Delta H_C$, then it may simply add terms to the equivalent of the higher-order equations (4.11) and (4.12) in the power series solution of equation (5.19). However, a problem arises if $H(x)$ has no constant-valued part H_0, but only higher-order parts. This will cause the Q-based term to vanish from the tracking problem equivalent of the first equation in the power series solution (4.10), which results in only a trivial solution to that equation and causes difficulties in the higher-order equations as well. This makes finding a usable feedback control with this method very difficult for that case. However, the linear tracking signal is sufficient for a large number of problems, including the HPCVD film thickness tracking we will do in Chapter 6, so we will consider only that form of the problem for now.

5.4 State Estimation for Nonlinear Systems With Nonlinear Measurements

The method for state estimation which we will construct here is related to the linear system case [57, 58] and to previous work on nonlinear systems in the literature [91, 92, 93, 94]. We consider a system

of ODEs with nonlinear dynamics and a nonlinear measurement of the form

$$\begin{cases} \dot{x}(t) &= f(x(t)) + Bu(x_e(t), t) \\ z(t) &= c(x(t)). \end{cases}$$

The feedback control for a tracking problem is given by $u(x_e, t) = -R^{-1}B^T \left[\Pi\left(x_e\right) x_e + s(t, x_{nom})\right]$ as discussed in the last section, except now in terms of the estimated state x_e.

We look for a state estimator of type $\dot{x}_e = f_c(x_e, t) + F(z, x_e)$. If the error in the estimation is $e = x - x_e$, then

$$\dot{e} = \dot{x} - \dot{x}_e = f(x) + Bu(x_e, t) - f_c(x_e, t) - F(c(x), x_e).$$

Let $f_c(x_e, t) = f(x_e) + Bu(x_e, t) - F(c(x_e), x_e)$. This leads to

$$\dot{e} = \left[f(x) - F(c(x), x_e)\right] - \left[f(x_e) - F(c(x_e), x_e)\right]. \tag{5.20}$$

As an indication of a good estimator, we want the error to be asymptotically stable, so that $e(t) \to 0$ (or $x_e(t) \to x(t)$) as $t \to \infty$. We want to choose a function F which will satisfy this condition.

For a linear problem, satisfying the stability condition is fairly straightforward, as described in [57, 58]. The dynamics in this case become $f(x) = A_0 x$, the measurement becomes $c(x) = C_0 x$ and we write the unknown function $F(z, x_e) = L_0 z = L_0 C_0 x$, so that we are seeking a constant-valued gain matrix L_0. The problem then reduces to $\dot{e} = (A_0 - L_0 C_0)e$. This is asymptotically stable when all the eigenvalues of $A_0 - L_0 C_0$ have negative real parts, so an L_0 must be found which results in such eigenvalues. Such an L_0 is guaranteed to exist if the pair (A_0, C_0) is observable.

An "optimal" choice of L_0 can be made by considering this problem as an optimal feedback control problem. Since the eigenvalues of a matrix are the same as those of its transpose, we can change the problem into that of forcing the eigenvalues of $\left(A_0 - L_0 C_0\right)^T = A_0^T - C_0^T L_0^T$ to have negative real parts. However, we note that in this form the system is related to the feedback control problem with $\tilde{A} = A_0^T$, $\tilde{B} = C_0^T$ and $\tilde{L} = L_0^T$ consisting of the system and cost functional

$$\begin{aligned} \dot{\tilde{x}} &= \tilde{A}\tilde{x} + \tilde{B}\tilde{u}, \\ \tilde{J} &= \frac{1}{2}\int_0^\infty \left(\tilde{x}^T U \tilde{x} + \tilde{u}^T V \tilde{u}\right) dt, \end{aligned}$$

with a control $\tilde{u} = -\tilde{L}\tilde{x}$. The optimal feedback gain for this problem, which best stabilizes the system given the cost functional, is given by $\tilde{L} = V^{-1}\tilde{B}^T\Sigma$, with Σ solving the Riccati equation

$$\Sigma\tilde{A} + \tilde{A}^T\Sigma - \Sigma\tilde{B}V^{-1}\tilde{B}^T\Sigma + U = 0.$$

Rewritten in terms of the original variables this is $L_0 = \Sigma C_0^T V^{-1}$ and

$$\Sigma A_0^T + A_0 \Sigma - \Sigma C_0^T V^{-1} C_0 \Sigma + U = 0.$$

This yields the "optimal" state estimation gain matrix L_0 for constructing the state estimator

$$\dot{x}_e = A_0 x_e + Bu(x_e, t) + L_0 \left(z - C_0 x_e\right).$$

Several papers in the literature have expanded on this state estimation formula to include nonlinear problems. Thau [91] considers systems with the linear and nonlinear parts split, such as

$$\dot{x} = A_0 x + g(x) + Bu, \tag{5.21}$$

where $g(x)$ contains only second-order and higher terms, and there is a linear measurement function $z = C_0 x$. Thau's estimator is of the form

$$\dot{x}_e = A_0 x_e + g(x_e) + Bu + L_0 \left(z - C_0 x_e\right),$$

with the gain matrix L_0 calculated so that the eigenvalues of the linear part of the problem, given by $A_0 - L_0 C_0$, have negative real parts. This can be done with a constant-valued Riccati equation in the manner described earlier. This leads to the following error equation:

$$\dot{e} = (A_0 - L_0 C_0) e + g(x) - g(x_e). \tag{5.22}$$

If $g(x)$ is locally Lipschitz and (A_0, C_0) is observable, this estimator will locally converge asymptotically. This can be shown as follows. The solution to (5.22) is given by

$$e(t) = \exp\left[(A_0 - L_0 C_0)t\right] e(0) + \int_0^t \exp\left[(A_0 - L_0 C_0)(t-s)\right] \left[g(x(s)) - g(x_e(s))\right] ds.$$

Given the observability of (A_0, C_0), we can set L_0 so that all the eigenvalues of $A_0 - L_0 C_0$ satisfy $\lambda < -b$ for some constant $b > 0$. We also assume a Lipschitz condition of $\|g(x) - g(y)\| < K \|x - y\|$, with $K < b$, for all x, y in some region Ω. Then the norm of the error is given by

$$\begin{aligned}
\|e(t)\| &\leq \|\exp\left[(A_0 - L_0 C_0)t\right] e(0)\| + \int_0^t \|\exp\left[(A_0 - L_0 C_0)(t-s)\right] \left[g(x(s)) - g(x_e(s))\right]\| \, ds \\
&\leq \exp\left[-bt\right] \|e(0)\| + \int_0^t \exp\left[-b(t-s)\right] \|g(x(s)) - g(x_e(s))\| \, ds \\
&\leq \exp\left[-bt\right] \|e(0)\| + \int_0^t \exp\left[bs\right] \exp\left[-bt\right] K \|x(s) - x_e(s)\| \, ds.
\end{aligned}$$

This leads to

$$\exp\left[bt\right] \|e(t)\| \leq \|e(0)\| + \int_0^t K \exp\left[bs\right] \|e(s)\| \, ds,$$

which with Gronwall's Inequality gives us

$$\begin{aligned} \exp[bt]\,\|e(t)\| &\leq \|e(0)\|\exp[Kt], \\ \|e(t)\| &\leq \|e(0)\|\exp[-(b-K)t]. \end{aligned}$$

Thus $e \to 0$ as $t \to \infty$ (the error is locally asymptotically stable in the region Ω).

Kou, Elliott and Tarn [92] consider a problem with nonlinear dynamics and measurement function and present a condition on the constant gain matrix L_0 which guarantees that the estimator

$$\dot{x}_e = f(x_e) + Bu + L_0(z - c(x_e)) \tag{5.23}$$

is asymptotically stable and the error decreases exponentially. However, it is often very difficult or impossible to find an L_0 which satisfies this condition, and so this is usually not very practical. Mielczarski [93] and Hu [94] use estimators of the type in equation (5.23), finding L_0 by separating the dynamics (as well as the measurement) into linear and nonlinear parts like Thau did in equation (5.21), and using those linear parts to find the gain matrix. They do this separation by linearizing $f(x)$ and $c(x)$ about the origin or some other expansion point to obtain matrices A_0 and C_0 for the linear parts of the dynamics and measurement, and then finding the gain matrix L_0 which results in all eigenvalues of $A_0 - L_0C_0$ having negative real parts.

The method we will describe here extends this nonlinear estimator technique further in a somewhat different direction. Instead of using a completely linearized system for finding the gain matrix L_0, we will use a state-dependent Riccati equation to solve for the gain from the nonlinear system itself. Rewriting f and c into matrix multiplication form as $f(x) = A(x)x$ and $c(x) = C(x)x$, and replacing F by $F(c(x), x_e) = L(x_e)C(x)x$, we find that equation (5.20) becomes

$$\dot{e} = [A(x) - L(x_e)C(x)]\,x - [A(x_e) - L(x_e)C(x_e)]\,x_e.$$

We further manipulate this equation by adding and subtracting terms to change it into a form more like equation (5.22):

$$\begin{aligned} \dot{e} &= [A(x) - L(x_e)C(x)]\,x - [A(x_e) - L(x_e)C(x_e)]\,x_e + (1-1)\,[A(x_e) - L(x_e)C(x_e)]\,x \\ &= [A(x_e) - L(x_e)C(x_e)]\,e + [A(x) - A(x_e) - L(x_e)C(x) + L(x_e)C(x_e)]\,x. \end{aligned}$$

We seek a gain L such that $\dot{e} = [A(x_e) - L(x_e)C(x_e)]\,e$ is asymptotically stable, and assume that the remaining term is small (which is reasonable if x is small, or x and x_e are close and A and C satisfy certain regularity properties). In analogy to the linear case, the "optimal" state-dependent estimator gain will be computed from a state-dependent Riccati equation as outlined below.

As in the linear case, we consider the transpose of the system, $A^T(x) - C^T(x)L^T(x)$. From there we set up the related feedback control problem with $\tilde{A} = A^T$, $\tilde{B} = C^T$ and $\tilde{L} = L^T$:

$$\begin{aligned} \dot{\tilde{x}} &= \tilde{A}(\tilde{x})\tilde{x} + \tilde{B}(\tilde{x})\tilde{u} \\ \tilde{J} &= \frac{1}{2}\int_0^\infty \left(\tilde{x}^T U \tilde{x} + \tilde{u}^T V \tilde{u}\right) dt \\ \tilde{u} &= -\tilde{L}(\tilde{x})\tilde{x}. \end{aligned}$$

The optimal feedback gain is now given by $\tilde{L}(\tilde{x}) = V^{-1}\tilde{B}^T(\tilde{x})\Sigma(\tilde{x})$, with $\Sigma(\tilde{x})$ solving the state-dependent Riccati equation

$$\Sigma(\tilde{x})\tilde{A}(\tilde{x}) + \tilde{A}^T(\tilde{x})\Sigma(\tilde{x}) - \Sigma(\tilde{x})\tilde{B}(\tilde{x})V^{-1}\tilde{B}^T(\tilde{x})\Sigma(\tilde{x}) + U = 0,$$

or, rewritten in terms of the original variables, $L(x) = \Sigma(x)C^T(x)V^{-1}$ with $\Sigma(x)$ satisfying

$$\Sigma(x)A^T(x) + A(x)\Sigma(x) - \Sigma(x)C^T(x)V^{-1}C(x)\Sigma(x) + U = 0.$$

This fully state-dependent version of the Riccati equation can be very difficult to solve due to the $\Sigma(x)C^T(x)V^{-1}C(x)\Sigma(x)$ term, especially if the measurement function is complicated, as it is for the optical absorption measurement we will use in the HPCVD problem (see Section 6.3). That measurement has an exponential function of the state, making it impossible to write as $c(x) = C(x)x$, let alone solve the above equation. To deal with this, we will make a simplification, linearizing the measurement for the purposes of finding $L(x)$. We wish to keep intact as much of the nonlinear nature of the problem as possible, though, so while we remove the state-dependence of $C(x)$, we keep $A(x)$ intact in the Riccati equation. Having done this, we can use the SDRE solution method described in Section 4.3.2 to find the state estimation gain

$$L(x) = \Sigma(x)C_0^T V^{-1} \tag{5.24}$$

from the state-dependent Riccati equation

$$\Sigma(x)A^T(x) + A(x)\Sigma(x) - \Sigma(x)C_0^T V^{-1}C_0\Sigma(x) + U = 0 \tag{5.25}$$

with $C_0 = C(0)$. The nonlinear dynamics and measurement are also still intact in the main part of the estimator,

$$\dot{x}_e = f(x_e) + Bu(x_e, t) + L(x_e)\left(z - c(x_e)\right), \tag{5.26}$$

as they are in the methods of Thau and others described above. We do not have a proof of the asymptotic stability of the error in this estimator like that for the Thau estimator error in equation (5.22). The stability analysis for the nonlinear estimator above is one of the aspects of this material yet to be studied.

5.5 Application to Test Problems

5.5.1 Simple Example System

We test the tracking and state estimation methods discussed in Sections 5.3 and 5.4 first on a simple example problem from [78], which was used earlier as a test problem for the nonlinear control methods in Chapter 4. The nonlinear control system is given by

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ x_1^2 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$$

in a factorized form appropriate for the SDRE. The cost functional is given by

$$J(x_0, u) = \frac{1}{2} \int_0^\infty \left(x^T Q x + u^T R u \right) dt,$$

or, slightly modified for the tracking problem,

$$J(x_0, u) = \frac{1}{2} \int_0^\infty \left[(y - r)^T Q (y - r) + u^T R u \right] dt.$$

First we will consider just the feedback tracking control problem, with no state estimation involved. We will track the variable $y = x_1$, attempting to force it to jump up from 0 to 0.5, hold, and then jump back to 0 (as shown in Figure 5.1). The weights in the cost function will be set to $Q = 10$ and $R = 1$, and the initial condition will be at the origin. The numerical computation of the controls and solution of the systems are done with MATLAB codes written by the author, using the built-in functions "*are*" for solving constant-valued algebraic Riccati equations and "*ode45*" for solving ODE systems. The two-point boundary value problem for the tracking variable in the nonlinear case is solved with a final time of $T_f = 15$, and 100 time discretization intervals. The first five terms of the SDRE power series solution are used.

The results of the tracking control problem are plotted in Figure 5.1 for the nonlinear tracking control described in equations (5.16)-(5.18), as well as a linear tracking control obtained by linearizing the problem and then using the standard linear techniques. The nonlinear control is obviously superior here, as the linearly controlled signal drastically overshoots the maximum of the desired trajectory. Raising the cost functional weight on the state to $Q = 100$ brings the linearly controlled signal down much closer to the nonlinear case, as shown in Figure 5.2, but the nonlinear control still produces better results.

Next, looking at a control problem with state estimation and without any tracking component, we will set the initial condition to $x_0 = [1, 0]^T$ and ask the control to force the system to 0. The

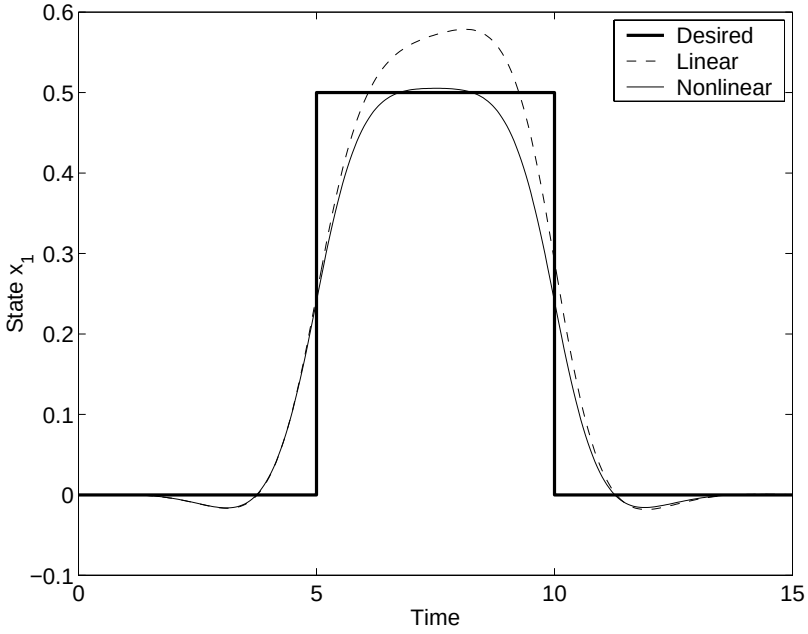

Figure 5.1: Comparison of feedback tracking controls on Example 1, with weight Q=10.

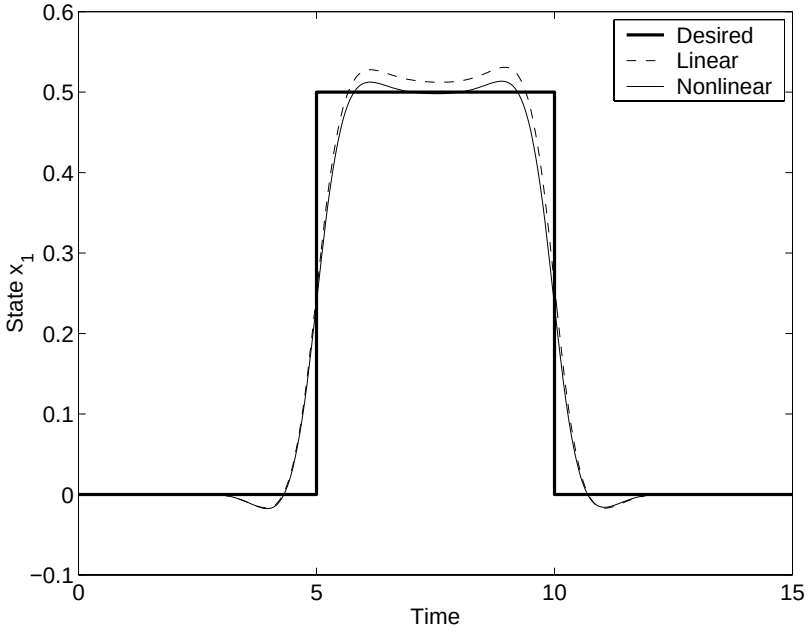

Figure 5.2: Comparison of feedback tracking controls on Example 1, with weight Q=100.

initial estimated state will be $(x_e)_0 = [1.3, 0]^T$, slightly off of the actual state, forcing the estimator to compensate. The estimation will be based on the nonlinear measurement $z = c(x) = x_1 + x_1 x_2$. In this problem we will set the cost functional weights to $Q = 10I_2$ (I_2 being a 2×2 identity matrix) and $R = 1$, with the weights in the state estimator gain problem being $U = I_2$ and $V = 1$. In Figure 5.3 we plot the actual and estimated states for two methods of state estimation described in Section 5.4. One is the method based on Thau's work, which is the nonlinear estimator in

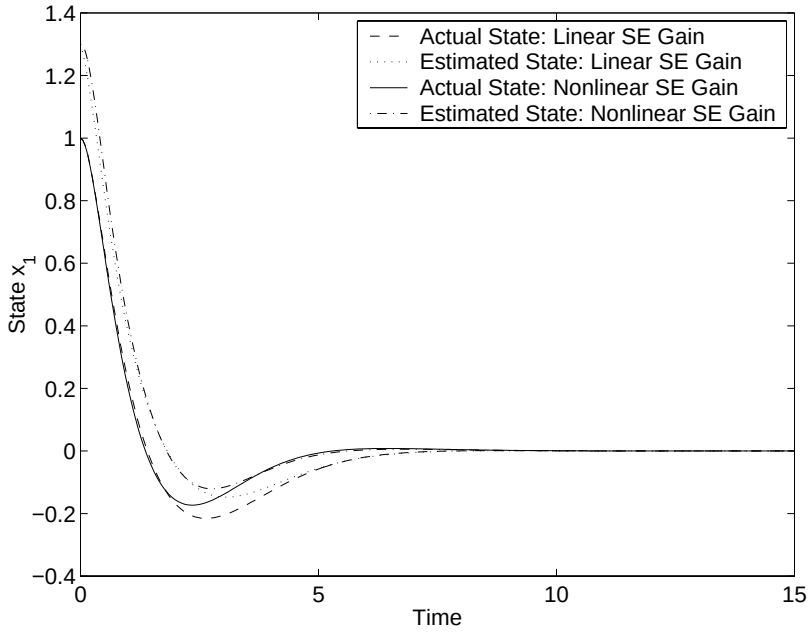

Figure 5.3: Actual and estimated states for feedback controls/state estimators in Example 1.

equation (5.23) using a linear gain found with a constant-valued Riccati equation (we will refer to this as the linear SE gain control). The other is the extension of this which is described in equations (5.24)-(5.26), a nonlinear estimator using a nonlinear gain from a state-dependent Riccati equation. The control using nonlinear SE gain performs slightly better than the linear SE gain control in this example, with the estimated state converging to the actual state faster and the state moving to 0 faster as well. A control found by completely linearizing the problem and finding a linear state estimator and control was tried but failed to yield state convergence in this case.

State estimation is now added to the tracking control problem, using the same desired step function as in Figures 5.1 and 5.2. The weights in this case are $Q = 10$, $R = 1$, $U = 10I_2$ and

$V = 1$, and the measurement is $z = x_1 + x_1 x_2$ as before. The initial condition is $x_0 = [0, 0]^T$ and the initial estimated state is $(x_e)_0 = [0.25, 0]^T$. Figure 5.4 depicts the actual and estimated states x_1 which result from the feedback control combining the nonlinear tracking and state estimation techniques. The estimator converges very nicely to the actual state while tracking the desired

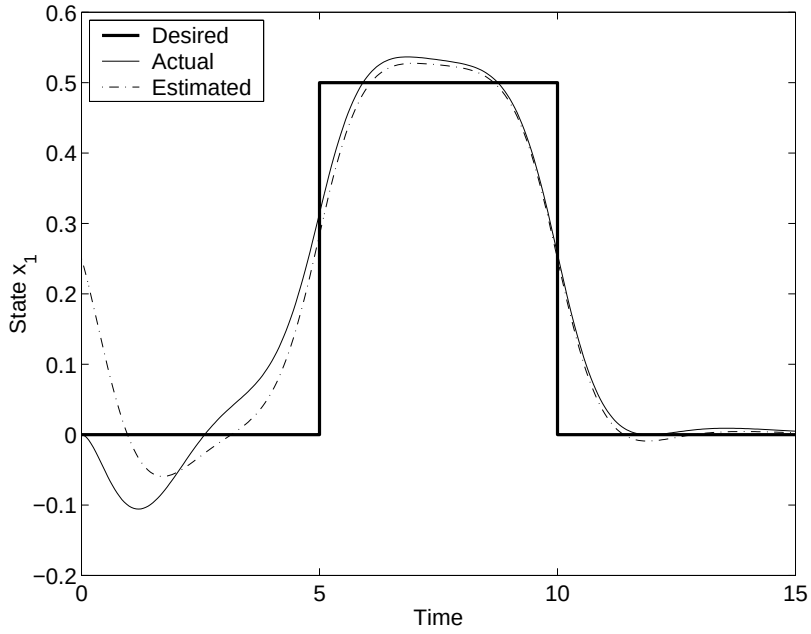

Figure 5.4: Actual and estimated states for nonlinear tracking control/estimator in Example 1.

trajectory fairly well. This fully nonlinear control is plotted, with a partially nonlinear control using the nonlinear tracking formula and the linear SE gain, and with the fully linear control found from the linearized system, in Figure 5.5. The fully nonlinear control performs better than the linear SE gain control at forcing the state to follow the desired trajectory (specifically in the latter half of the time period), while the fully linear control is far less effective than the other two.

Finally, we consider the same problem as in Figures 5.4 and 5.5, with both state estimation and tracking, except that instead of a deviation in the initial estimated state for which the estimator must compensate, there is instead added random noise in the problem. The noise consists of independent uniform distributions $\varepsilon_1(t)$ and $\varepsilon_2(t)$, with $|\varepsilon_k| \leq 0.1$ (20% of the maximum desired x_1 trajectory),

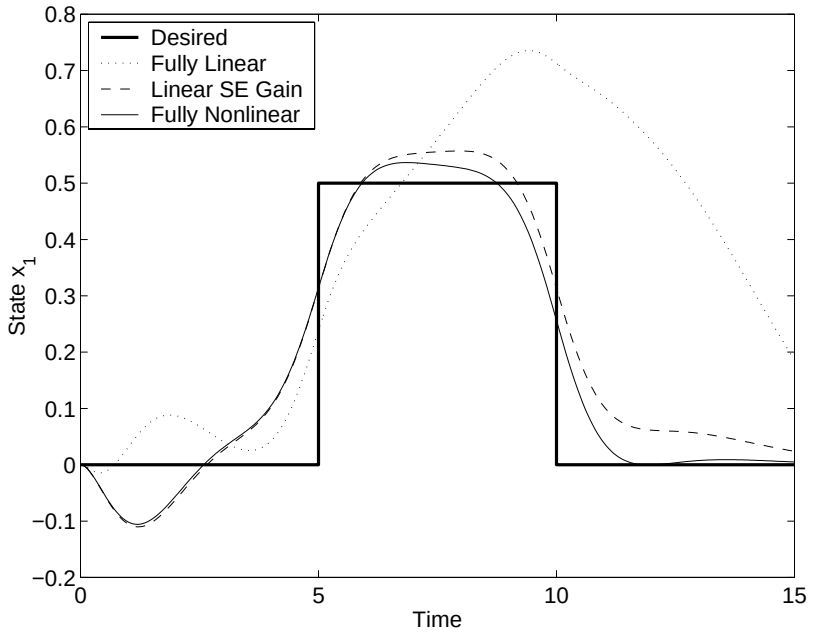

Figure 5.5: Comparison of tracking controls/state estimators on Example 1, with bad $(x_e)_0$.

one added to the dynamics and the other to the measurement:

$$\begin{cases} \dot{x} &= A(x)x + Bu + \varepsilon_1 \\ z &= x_1 + x_1 x_2 + \varepsilon_2. \end{cases}$$

In Figure 5.6 we compare the results of the fully nonlinear, linear SE gain, and fully linear controls applied to this problem. The fully nonlinear and the linear SE gain controls yield very similar results here, and as in Figure 5.5 both perform much better than the fully linear control.

5.5.2 Flight Dynamics Example System

The second example to which we apply the state estimation and tracking control methods from Sections 5.3 and 5.4 is a modified version of the flight dynamics example from [61], which was also used as test problem in Chapter 4. The system is given by

$$\dot{x} = (A_0 + x_2 A_{NL})\, x + Bu,$$

with the matrices A_0, A_{NL} and B as in Section 4.4.3. The cost functional to be minimized is

$$J(x_0, u) = \frac{1}{2} \int_0^\infty \left[(y-r)^T Q (y-r) + u^T R u \right] dt.$$

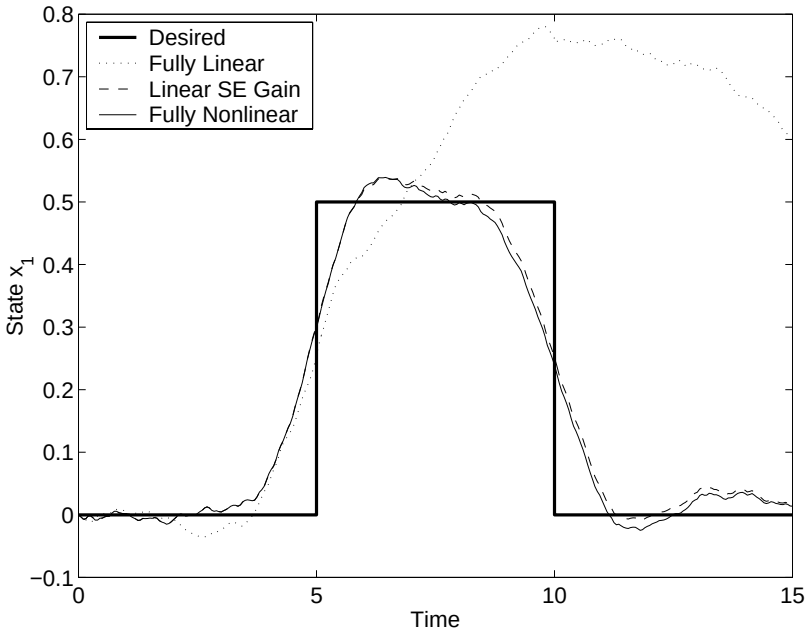

Figure 5.6: Comparison of tracking controls/state estimators on Example 1, with noise.

when in tracking problem form. The five state variables in this model represent the flight conditions of the aircraft, as described in Section 4.4.3. The control u is the input deflection to the canard flaps in radians. The model is explained in much more detail in Section 4.4.3 and in [61].

In the first test we consider a control problem with state estimation but no tracking. The feedback control will attempt to force the state variables to 0 from an initially large angle of attack x_2, given by an initial condition of $x_0 = [0, 15(\pi/180), 0, 0, 0]^T$. However, the initial estimated state is $(x_e)_0 = [0, 20(\pi/180), 0, 0, 0]^T$. The measurement for the estimator is of the velocity and the canard deflection, so that $z = c(x) = [x_1, x_5]^T$. The cost functional weights are given by $Q = I_5$ and $R = 100$, and the weights for the estimator gain problem are $U = 100I_5$ and $V = I_2$. Figure 5.7 depicts the actual and estimated states for this problem using the linear SE gain (Thau) control and the nonlinear SE gain control found with an SDRE solution. For each method it takes some time for the estimated state to converge to the actual state, and for both to be forced to 0, but they both do so in a smooth manner. There is a larger oscillation noticeable in the linear SE gain control, causing slower convergence. On the other hand, with the fully linear state estimation and control algorithm applied the system remains unstable and does not converge.

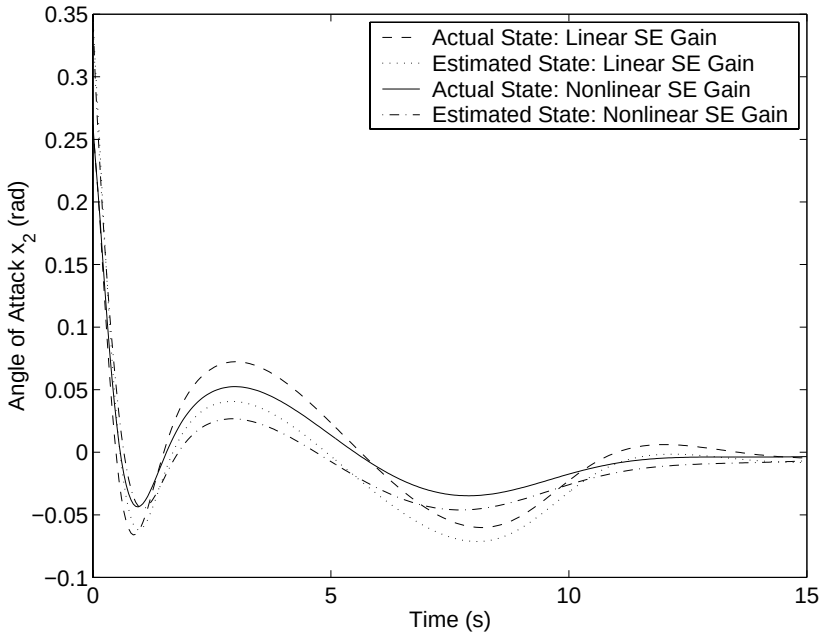

Figure 5.7: Actual and estimated states for feedback controls/state estimators in Example 2.

Next we will consider a problem with tracking but no state estimation. The objective is to track a desired flight path angle x_4, increasing it gradually from 0 up to $45(\pi/180)$, holding, and then returning to 0 (as shown in Figure 5.8). The initial condition is at the origin, and the cost weights are $Q = 1$ and $R = 10$. The results are plotted in Figure 5.8 for the nonlinear and linear tracking controls. Here the nonlinear control does not perform substantially better than the linear control; in fact the linear control yields better results at the top of the ramp up to $45(\pi/180)$, though late in the time period the linear control returns to 0 more slowly than the nonlinear.

In considering a problem with both tracking and state estimation, we use the same desired trajectory as before, forcing the flight path angle x_4 from 0 up to $45(\pi/180)$ and back. Estimation is added using a measurement of the velocity, angle of attack, and canard deflection: $z = [x_1, x_2, x_5]^T$. The weights are $Q = 1$, $R = 1$, $U = 100I_5$ and $V = I_3$. The actual state starts at the origin, as in the previous tracking problem, but the estimated state starts slightly off of the actual state, at $(x_e)_0 = [0, 0, 0, 5(\pi/180), 0]^T$. Figure 5.9 depicts the estimated state almost converging to the actual state by the time of the desired x_4 increase, and remaining close to the actual state for the rest of the time period. In Figure 5.10 we plot the actual state when controlled using our fully

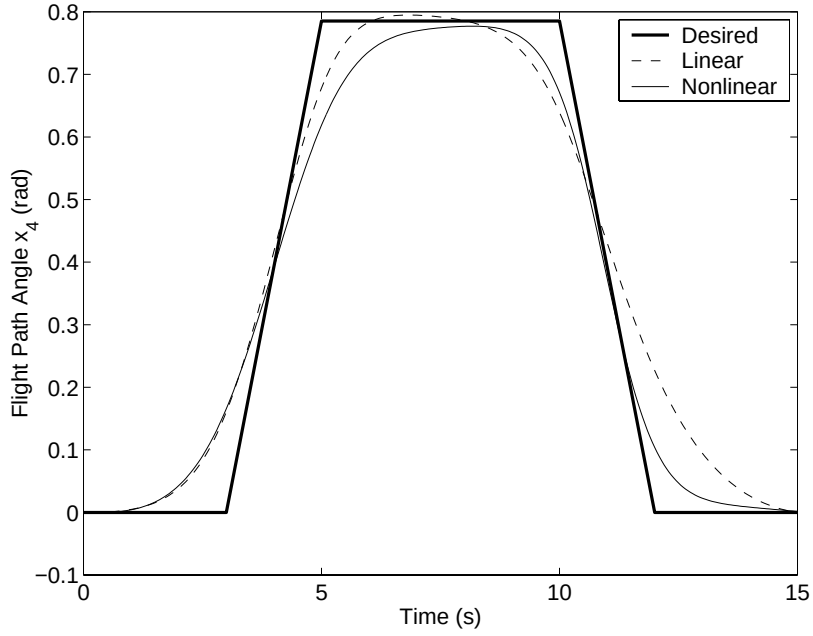

Figure 5.8: Comparison of feedback tracking controls on Example 2.

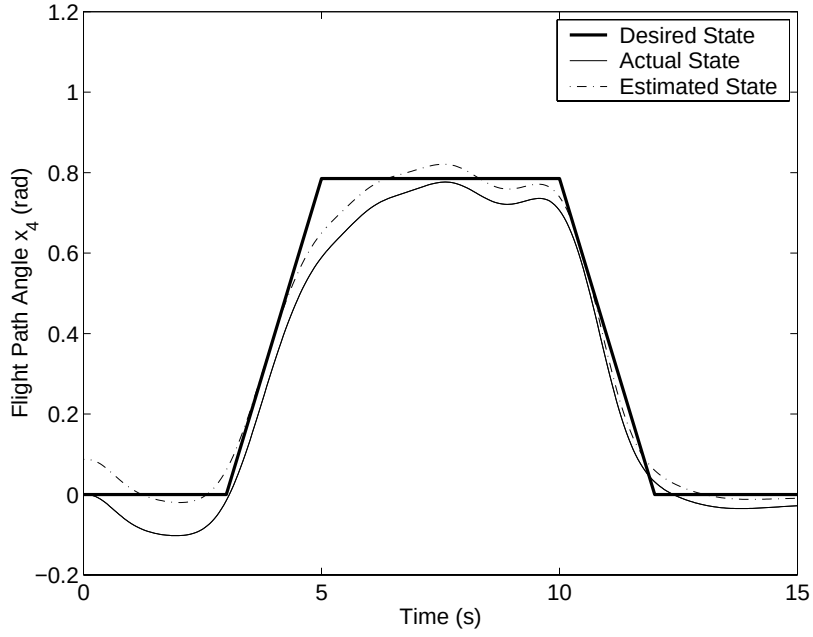

Figure 5.9: Actual and estimated states for nonlinear tracking control/estimator in Example 2.

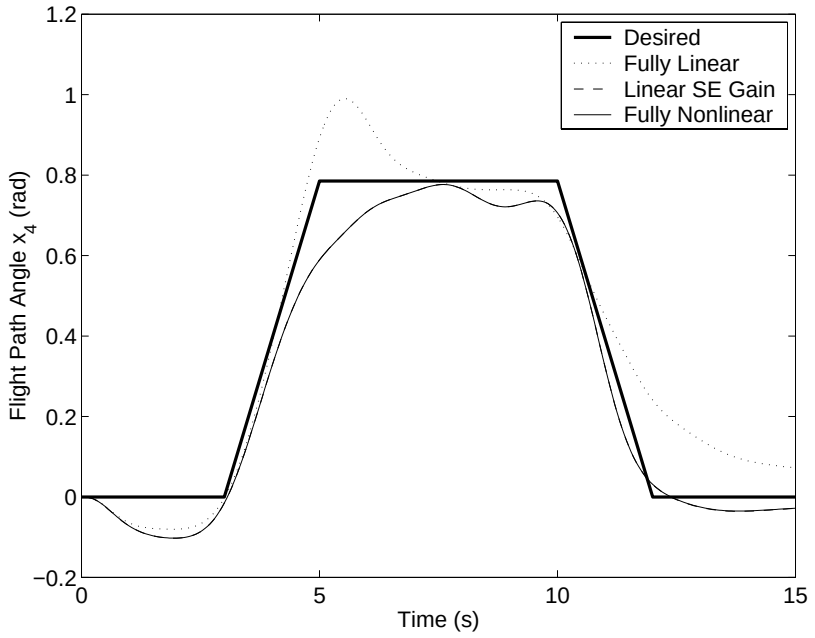

Figure 5.10: Comparison of tracking controls/state estimators on Example 2, with bad $(x_e)_0$.

nonlinear algorithm, as well as when using the linear SE gain control, and the fully linear control. It can be seen that the linear control overshoots significantly at the top of the ascent and is very slow to return to 0. The other two methods produce virtually identical results, with the difference indiscernible in the plots in Figure 5.10.

Finally, we alter the problem used in Figures 5.9 and 5.10, involving both state estimation and tracking, by removing the deviation in the initial estimated state and adding random noise to the problem. The independent uniform distributions $\varepsilon_1(t)$ and $\varepsilon_2(t)$, with $|\varepsilon_k| \leq 2.25(\pi/180)$ (5% of the maximum desired flight path angle, and an even larger percentage of the actual state variables at most times), are added to the dynamics and the measurement respectively:

$$\begin{cases} \dot{x} &= (A_0 + x_2 A_{NL})\, x + Bu + \varepsilon_1 \\ z &= [x_1, x_2, x_5]^T + \varepsilon_2. \end{cases}$$

The results of the fully nonlinear, linear SE gain, and fully linear controls applied to this problem are plotted in Figure 5.11, where it can be seen that the two nonlinear controls are again very similar, and that both perform better than the linear control at tracking the desired trajectory both at its maximum and as it returns to 0.

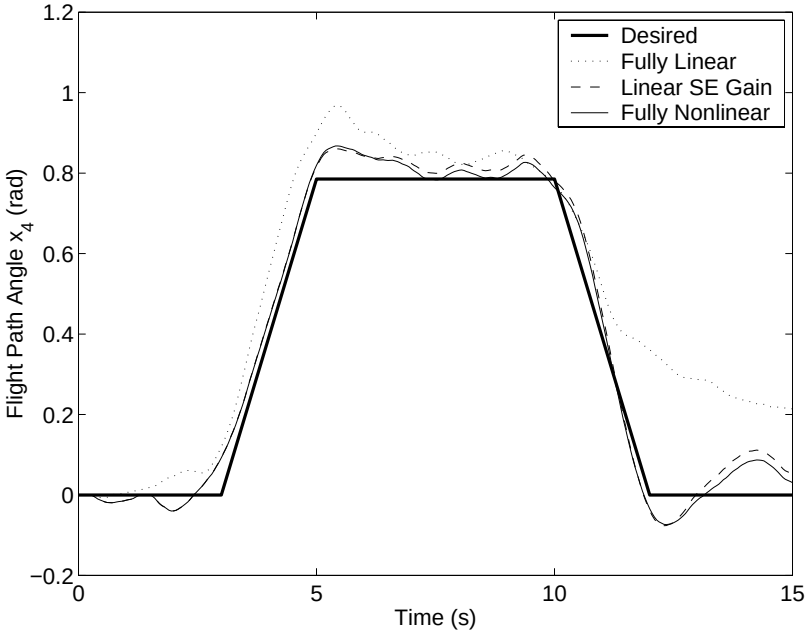

Figure 5.11: Comparison of tracking controls/state estimators on Example 2, with noise.

5.6 Conclusions

In this chapter we have considered the method for feedback control of nonlinear systems using the state-dependent Riccati equation and extended it into a feedback tracking control method. We have also modified the state estimation method for nonlinear systems established in the literature to include a nonlinear gain function found through a state-dependent Riccati equation. Application of these new techniques to two selected example problems provided significant control authority and distinct advantages in comparison with the linear methods.

As mentioned earlier in this chapter, there are some restrictions and drawbacks to the new techniques which must be considered. The power series solution of the SDRE method grows inaccurate when the states move further from the origin, something which is of particular concern in a tracking problem. In solving the tracking variable equation by coupling it with a nominal state equation, one tacitly assumes a good prediction of the actual state behavior for the control to be effective. There are limitations on the types of problems to which the SDRE approach can be applied and on the types of signals which can be tracked, and the SDRE for obtaining the nonlinear state estimation gain uses only a linearized version of the measurement function.

While these are nontrivial factors to consider, the methods described here for tracking control and state estimation are still applicable to a large class of important control problems, and their performance on the chosen examples provides improvement (in places dramatic improvement) when compared to previously established control techniques. In Chapter 6 we will apply the nonlinear tracking control and estimation methods to the reduced order HPCVD system combining the gas-phase and surface kinetics models constructed earlier.

Chapter 6

Surface Flux Tracking With State Estimation

6.1 Introduction

In this chapter we will put together the material of the previous chapters to construct the full model of the HPCVD film growth process including the gas-phase POD model from Chapter 3 and the ROSK surface model from Chapter 2. The objective will be to track the growing film thickness during a single growth cycle, using real-time measurements of the system to estimate the state and a feedback control based on this estimated state to guide the film thickness increase. Section 6.2 will describe some modifications to the ROSK model as presented in Chapter 2, and the creation of the complete reduced order model by coupling the gas-phase and surface portions of the model through the species flux at the surface. The state estimation is based on partial observations of the gas-phase system through a nonlinear optical absorption measurement which is described is Section 6.3. Section 6.4 will discuss the formulation of the tracking control and state estimation problem for the HPCVD reduced order model. Due to the nonlinear dynamics in the surface model we use the SDRE-based nonlinear control and estimation formulas from Chapter 5. The results of the control problem simulations are analyzed in Section 6.5, with some overall conclusions about the reduced order HPCVD control problem given in Section 6.6.

6.2 Linking the Gas-Phase and Surface Models

The complete HPCVD process model is found by linking the gas-phase reduced order model from Section 3.4, in equation (3.22), with a modified version of the reduced order surface kinetics (ROSK) model described in detail in Chapter 2, given by equations (2.4)- (2.7). Here, instead of pulsed

ballistic beams for the source terms in the surface equations, the single source term is based on the flux at the surface from the gallium-containing species in the gas phase. The gas-phase model is given by

$$\dot{y}^M(t) = A^M y^M(t) + B^M u(t), \tag{6.1}$$

in terms of the POD coefficients from the reduced order representation of the state in equation (3.21). The single control input u is the inlet mass fraction of Y_1 (TMG). From the species mass fractions Y_1, Y_2 and Y_3, representing TMG, DMG and MMG respectively, the flux of gallium to a specific point $\overrightarrow{x}_p$ at the center of one of the substrate surfaces is given by

$$q(\overrightarrow{Y}) = -\rho \left[D_1 \frac{W_{Ga}}{W_1} \frac{\partial Y_1}{\partial \overrightarrow{n}}\bigg|_{\overrightarrow{x}_p} + D_2 \frac{W_{Ga}}{W_2} \frac{\partial Y_2}{\partial \overrightarrow{n}}\bigg|_{\overrightarrow{x}_p} + D_3 \frac{W_{Ga}}{W_3} \frac{\partial Y_3}{\partial \overrightarrow{n}}\bigg|_{\overrightarrow{x}_p} \right].$$

Writing this gallium flux as an approximation in terms of the coefficients of the POD basis functions, we have

$$\begin{aligned} q^{M_G}(y^{M_G}(t)) &= -\rho \sum_{n=1}^{3} \left[D_n \frac{W_{Ga}}{W_n} \sum_{k=1}^{M_n} \frac{\partial \Phi_{nk}}{\partial \overrightarrow{n}}\bigg|_{\overrightarrow{x}_p} y_{nk}(t) \right] \\ &= H_q^{M_G} y^{M_G}(t). \end{aligned}$$

The molar weight values are $W_1 = 114.8$ g/mol, $W_2 = 99.79$ g/mol, $W_3 = 84.755$ g/mol, and $W_{Ga} = 69.9$ g/mol. The density ρ and diffusivities D_n are dependent on the temperature at the point $\overrightarrow{x}_p$ on the substrate, with the density value found through equation (3.5) and the diffusivity values interpolated from literature values [42]. This flux will be used to couple the gas phase state variables (the POD coefficients y_{nk}) to the surface model, which has species concentrations n_i as state variables. The size of the gas-phase model is now given by M_G instead of M as in Chapter 3; M will now refer to the size of the combined system.

The ROSK model describing the surface processes is somewhat altered from its version in Chapter 2. To begin with, the species represented by the variables n_i are different. Since we are using trimethylgallium (TMG) as the original gas-phase source vapor instead of triethylgallium (TEG) as previously, and there is substantial decomposition during the flow through the high-pressure reactor, we will now consider n_1, the intermediate stage of the gallium species decomposition, to represent the amount of gallium on the surface. The species n_2 will represent the amount of activated gallium (available to react, unlike n_1) on the surface, while n_3 will represent the amount of gallium phosphide in the film. In the simulations in this chapter we are only going to look at a single pulse, rather than the long-term behavior of the system, and we will consider the phosphorus source to

have already arrived at the surface. The initial concentration of activated phosphorus is chosen as an amount in excess of that needed to grow the desired GaP layer thickness, and is specifically given by $S_p = \gamma S_0/N_A$, where S_0 is the density of surface sites on the substrate, N_A is Avogadro's number, and $\gamma > 1$ is a chosen constant. The concentration of activated surface phosphorus at any time can be found from the initial amount on the surface less the amount incorporated into the film up to that time, as $S_p - n_3(t)$, thus removing the ODE representing phosphorus from the system. This serves to simplify the model, also making it more controllable, since the amount of active surface phosphorus would be very difficult to regulate without any TBP input control. The model is also simplified by removing the desorption loss terms, which should be less significant under the high surface pressure in the HPCVD reactor than in the near-vacuum reactor used in Chapter 2.

Another difference in the surface model used in this chapter is that, instead of the whole of the substrate, we are now looking at a particular point at the center of it, and basing the surface model on the incoming flux from the gas phase at that point. For this reason we change all the n_i variables to molar concentrations in units of mol/m^2, by dividing the original n_i variables (the total number of moles in the deposition region) by the substrate/susceptor area $A = 400\pi$ mm^2. The rate constant k_1 for transformation from gallium to activated gallium does not change as a result of this, but the rate constant k_{GaP} for formation of gallium phosphide will have to be multiplied by A in order to compensate for the fact that it is the coefficient of the product of two species. The scaling factor of $1/10^{-8}$ moles is also incorporated into k_{GaP} in this new model. Finally, the gas-phase flux is used as the source term for gallium in the surface reaction layer by the formula $S_1(t) = q^{M_G}(t)/W_{Ga}$. All the state variables in the surface model ODE system are assumed to be initially zero. With all these changes, the modified ROSK model is given by

$$\begin{aligned}
\dot{n}_1(t) &= \frac{q^{M_G}(t)}{W_{Ga}} - k_1 n_1(t) \\
\dot{n}_2(t) &= k_1 n_1(t) - k_{GaP}\left[S_p - n_3(t)\right] n_2(t) \\
\dot{n}_3(t) &= k_{GaP}\left[S_p - n_3(t)\right] n_2(t),
\end{aligned}$$

with n_1, n_2, and n_3 the moles per m^2 of Ga, activated Ga, and GaP respectively. For this modified ROSK model we have chosen the rate constants to be $k_1 = 20$ s^{-1} and $k_{GaP} = 2000$ m^2(mol·s); values of these parameters which better fit the physical surface processes would have to be found from experimental data as in Chapter 2, but these will serve for a proof-of-concept demonstration of the model and its control. The surface model has size $M_S = 3$ and can be written in vector form as $\dot{n}^{M_S} = f(n^{M_S}, q^{M_G})$. The combined gas-phase/surface system will have size $M = M_G + M_S$.

6.3 Optical Measurement in the HPCVD Reactor

The real-time observation of the gas-phase species transport system is done with a measurement of the intensity of a beam of light traveling across the width of the reactor above a substrate surface. The amount of light reaching the detector after absorption in the reactor depends on the gas-phase species mass fractions along the path of the light beam. The light source and light detection locations on the sides of the reactor for this horizontal observation technique are shown in Figure 6.1. The

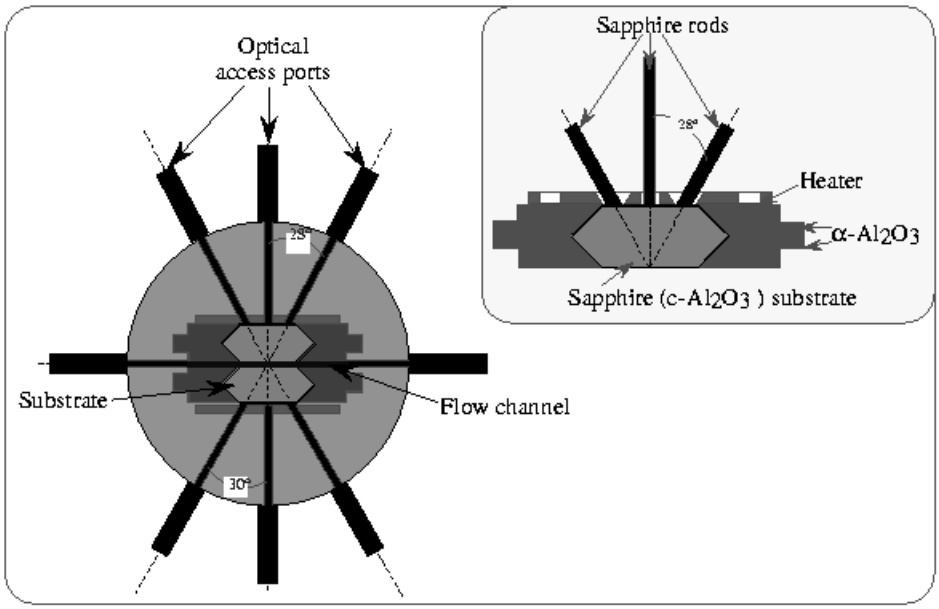

Figure 6.1: Measurement techniques in the HPCVD reactor.

optical access ports behind the substrates are for a variation of the PRS measurement, taken from the reverse side of the substrate, using the amount of light reflected back to obtain a partial measurement of the properties of the film and surface reaction layer on the substrate surface. Our model will not use this as an observation method, however, but will include only the optical absorption measurement of the gas-phase species. The optical access ports for both of these measurements can be seen in the photograph of the physical reactor in Figure 1.1.

The intensity of the light signal as measured at the detector side of the channel is

$$I = I_0 \exp\left(-\int_{\mathcal{W}} \frac{4\pi}{\lambda}\epsilon_{imag}(\overrightarrow{x})d\overrightarrow{x}\right), \tag{6.2}$$

where $\mathcal{W}$ represents the path across the 50 mm width of the reactor above the target point $\overrightarrow{x}_p$ at the center of one of the substrates. The wavelength of the light is λ, and I_0 and I are the initial and detected intensities respectively. The parameter $\epsilon_{imag}(\overrightarrow{x})$ is the imaginary component of the dielectric function of the gas mixture at a location $\overrightarrow{x}$ in the reactor. This dielectric function is given by

$$\begin{aligned} \epsilon(\overrightarrow{x}) &= \epsilon_{real}(\overrightarrow{x}) - i\epsilon_{imag}(\overrightarrow{x}) \\ &= 1 + \left(\sum_{k=0}^{N} \frac{Y_k(\overrightarrow{x})}{W_k} \right)^{-1} \sum_{n=0}^{N} \frac{Y_n(\overrightarrow{x}) F_n}{W_n}, \end{aligned}$$

where N is the number of gas-phase species ($N = 3$ in our model) and species $n = 0$ represents the carrier gas. Y_n is the mass fraction of species n in the gas-phase model described in Chapter 3, and W_n is the molar weight of species n. The parameter $F_n = a_n - ib_n$ is the complex optical response of species n.

The wavelength λ is typically chosen so that the absorption measurement is particularly sensitive to one species. This is achieved through using a wavelength at which the imaginary part of the optical response of that particular species is larger than those of the other species. Since experimental data is not yet available, we use the following assumed values for the sake of implementation: $b_0 = 0$, $b_1 = 5 \times 10^{-3}$, and $b_2 = b_3 = 5 \times 10^{-4}$, all at $\lambda = 1000$ nm. With these values and the assumption of a dominant carrier gas due to the dilute approximation, the dielectric function imaginary part is given by

$$\epsilon_{imag}(\overrightarrow{x}) = W_0 \sum_{n=1}^{3} \frac{b_n Y_n(\overrightarrow{x})}{W_n}. \tag{6.3}$$

The molar weight values are $W_0 = 28.01$ g/mol, $W_1 = 114.8$ g/mol, $W_2 = 99.79$ g/mol, and $W_3 = 84.755$ g/mol. To normalize the observed signal described by equations (6.2) and (6.3), we consider the actual observation, which will be used in the state estimation process described in the next section, to be

$$\begin{aligned} z(\overrightarrow{Y}) &= 1 - \frac{I}{I_0} \\ &= 1 - \exp\left(-\frac{4\pi W_0}{\lambda} \int_{\mathcal{W}} \sum_{n=1}^{3} \frac{b_n}{W_n} Y_n(\overrightarrow{x}) d\overrightarrow{x} \right). \end{aligned} \tag{6.4}$$

6.4 Constructing the Control Problem

The combined system consisting of the gas phase model found in Section 3.4 linked to the modified ROSK model, as described in Section 6.2, is given by

$$\begin{bmatrix} \dot{y}^{M_G}(t) \\ \dot{n}^{M_S}(t) \end{bmatrix} = \begin{bmatrix} A^{M_G} y^{M_G}(t) \\ f\left(n^{M_S}(t), q^{M_G}(t)\right) \end{bmatrix} + \begin{bmatrix} B^{M_G} \\ 0 \end{bmatrix} u(t).$$

This can be written in terms of a single state vector as

$$\dot{x}^M(t) = \mathcal{A}(x^M(t))x^M(t) + \mathcal{B}u(t), \tag{6.5}$$

with $\mathcal{A}(x^M)$ an $M \times M$ state-dependent matrix function of the combined state x^M, $\mathcal{B}$ a length M constant-valued vector, and u the single control input (the inlet TMG mass fraction). The initial state x_0^M is zero for all the gas-phase coefficients as well as all the surface species. The signal we will track is the gallium phosphide film thickness

$$\begin{aligned} d^M(t) &= V_{GaP} n_3(t) \\ &= H^M x^M(t), \end{aligned} \tag{6.6}$$

where $V_{GaP} = 12.2$ cm^3/mol is the molar volume of GaP.

To construct the tracking control problem for the combined system (6.5) and tracking signal (6.6) we start with the cost functional

$$J(x_0^M, u) = \frac{1}{2}\int_0^\infty \left[(d^M - d_T)^T Q_1 (d^M - d_T) + (\overline{d}^M)^T Q_2 \overline{d}^M + u^T R u\right] dt. \tag{6.7}$$

The weights are $R = 1$, $Q_1 = r_1$, and $Q_2 = r_2 I_M$, with I_M the $M \times M$ identity matrix and r_1 and r_2 two chosen design parameters. The desired film thickness trajectory $d_T(t)$ is an upward ramp of growth from 0 to the final thickness of 1 monolayer of GaP (roughly 10^{-11} m), with the ends of the ramp somewhat rounded off where the growth process starts and stops (see Figure 6.2). The time axis in this figure has been nondimensionalized as well; each time unit corresponds to 0.1 s.

Note that the cost functional (6.7) has an additional term $(\overline{d}^M)^T Q_2 \overline{d}^M$ which was not included in the tracking control development in Section 5.3. This is added because the tracking signal $d^M = H^M x^M$ is of lesser rank (here rank 1) than the system rank M. It is reasonable to attempt to increase the number of constraints in the problem to M, without creating a conflict with the original tracking signal constraint $(d^M - d_T)^T Q_1 (d^M - d_T)$. The additional variables are found as follows: $\overline{d}^M = \overline{H}^M x^M$, with $\overline{H}^M = I_M - L^M H^M$ and $L^M = (H^M)^T \left[H^M (H^M)^T\right]^{-1}$. The

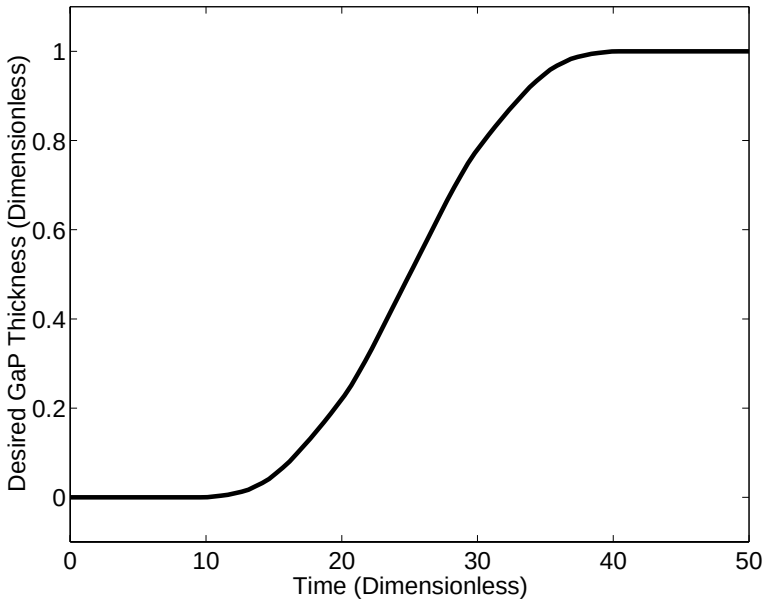

Figure 6.2: Desired film thickness growth profile.

new term encourages the control to drive the parts of the system independent of d^M to zero, thus keeping the rest of the system from diverging while it tries to force d^M to follow the desired tracking trajectory. This results in a total of M constraints on the M-variable state (see pp. 69-70 in [57] for more information). This added cost term was successfully implemented in [6, 7], and we will use it in our control formulation as well, in the hope that it will minimize any unusual behavior in the system which is not necessary to achieve the desired gallium flux.

This problem contains the nonlinear dynamics of the surface reactions, so we will be using the nonlinear SDRE-based feedback tracking control method from Section 5.3. The optimal control is given by

$$u(x^M, t) = -R^{-1}\mathcal{B}^T \left[\Pi(x^M)x^M + s(t, x^M_{nom})\right], \tag{6.8}$$

where $\Pi(x^M)$ is the solution to the state-dependent Riccati equation

$$\Pi(x^M)\mathcal{A}(x^M)+\mathcal{A}^T(x^M)\Pi(x^M)-\Pi(x^M)\mathcal{B}R^{-1}\mathcal{B}^T\Pi(x^M)+(H^M)^TQ_1H^M+(\overline{H}^M)^TQ_2\overline{H}^M = 0. \tag{6.9}$$

The added state cost term in the cost functional (6.7) results in the term $(\overline{H}^M)^TQ_2\overline{H}^M$ being added into the above Riccati equation. This SDRE is solved using the power series approximation described in Section 4.3.2, beginning by rewriting the state dynamics matrix as a sum of constant

and state-dependent parts by $\mathcal{A}(x^M) = A_0 + \varepsilon g(x^M)\Delta A_C$ with a temporary variable ε. The SDRE solution $\Pi(x^M)$ is expanded as a power series in ε, resulting in

$$\Pi(x^M, \varepsilon) = \sum_{n=0}^{\infty} \varepsilon^n g^n(x^M)(L_n)_C. \tag{6.10}$$

Substituting these two expansions into equation (6.9) and matching terms with the same powers of ε results in the following set of equations for determining the constant-valued matrices $(L_n)_C$:

$$\begin{aligned}
(L_0)_C A_0 + A_0^T (L_0)_C - (L_0)_C \mathcal{B} R^{-1} \mathcal{B}^T (L_0)_C + (H^M)^T Q_1 H^M + (\overline{H}^M)^T Q_2 \overline{H}^M &= 0, \quad (6.11)\\
(L_1)_C \left[A_0 - \mathcal{B} R^{-1} \mathcal{B}^T (L_0)_C\right] + \left[A_0^T - (L_0)_C \mathcal{B} R^{-1} \mathcal{B}^T\right](L_1)_C + (L_0)_C \Delta A_C & \\
+ \Delta A_C^T (L_0)_C &= 0, \quad (6.12)\\
(L_n)_C \left[A_0 - \mathcal{B} R^{-1} \mathcal{B}^T (L_0)_C\right] + \left[A_0^T - (L_0)_C \mathcal{B} R^{-1} \mathcal{B}^T\right](L_n)_C + (L_{n-1})_C \Delta A_C & \\
+ \Delta A_C^T (L_{n-1})_C - \sum_{k=1}^{n-1} \left[(L_k)_C \mathcal{B} R^{-1} \mathcal{B}^T (L_{n-k})_C\right] &= 0. \quad (6.13)
\end{aligned}$$

This series of equations can be solved easily for as many terms of the power series as desired, before finding the SDRE solution by substituting them back into (6.10) with ε set to 1. In our HPCVD problem the separated parts of $\mathcal{A}(x^M)$ are

$$A_0 = \begin{bmatrix} A^{M_G} & 0^{M_G \times M_S} \\ H_q^{M_G}/W_{Ga} & -k_1 & 0 & 0 \\ 0^{1\times M_G} & k_1 & -k_{GaP} S_p & 0 \\ 0^{1\times M_G} & 0 & k_{GaP} S_p & 0 \end{bmatrix},$$

$$\Delta A_C = \begin{bmatrix} 0^{M_G \times M_G} & 0^{M_G \times M_S} \\ & 0 & 0 & 0 \\ 0^{M_S \times M_G} & 0 & 0 & k_{GaP} \\ & 0 & 0 & -k_{GaP} \end{bmatrix},$$

with $g(x^M) = n_2$. In the simulations that follow we use $N_P = 5$ power series terms in the expansion for $\Pi(x^M)$ in equation (6.10).

The other part of the control formulation is the tracking variable $s(t, x^M_{nom})$, which is found using the two-point boundary value problem technique developed in Section 5.3. This is unchanged by the addition of the Q_2 term to the cost functional, since the added variables are not tracking a nonzero trajectory but are simply being stabilized to zero. The tracking variable and a nominal

state variable form the coupled system

$$\begin{cases} \dot{s} &= -\mathcal{A}^T(x^M_{nom})s + \Pi\left(x^M_{nom}\right)\mathcal{B}R^{-1}\mathcal{B}^T s + (H^M)^T Q_1 d_T - D_t\Pi\left(x^M_{nom}\right)x^M_{nom} \\ & \quad - \sum_{i=1}^{M}\left(x^M_{nom}\right)_i\left(\frac{\partial \mathcal{A}_{1\to M,i}}{\partial x^M_{nom}}(x^M_{nom})\right)^T\left(\Pi\left(x^M_{nom}\right)x^M_{nom} + s\right) \\ \dot{x}^M_{nom} &= \mathcal{A}(x^M_{nom})x^M_{nom} - \mathcal{B}R^{-1}\mathcal{B}^T(\Pi\left(x^M_{nom}\right)x^M_{nom} + s), \end{cases} \tag{6.14}$$

with initial condition $x^M_{nom}(0) = x^M_0$ and final condition $s(T_f, x^M_{nom}(T_f)) = 0$. The final time used in the two-point boundary value problem is given by the sum $T_f = T_p + \Delta$. Here T_p is the ending time of the studied target thickness trajectory, and Δ is an additional time value to allow this problem to approximate the infinite-time problem well and to be stabilized completely by the control. In a linear problem Δ can be found as five times the dominant time constant associated with the eigenvalues of the matrix $\mathcal{A} - \mathcal{B}R^{-1}\mathcal{B}^T\Pi$ (see p. 86 of [57]). For the nonlinear HPCVD problem, the eigenvalues of the linearized matrix $A_0 - \mathcal{B}R^{-1}\mathcal{B}^T(L_0)_C$ could be used. However, in addition to the nonlinearity, our desired tracking signal is a ramp function which is not zero at the target trajectory final time $T_p = 50$ (in dimensionless time units), which causes problems with forcing s to return to zero at only a short time after T_p. Thus we use a large value ($\Delta = 50$) to avoid this, although we are only concerned with the system behavior up to T_p.

The coupled system (6.14) is discretized and solved numerically using the mixed finite difference method described in Section 5.3 to obtain the tracking variable $s(t, x^M_{nom})$. The tracking variable, found offline, is then combined with the direct state feedback term involving the state-dependent Riccati equation solution in the feedback control formulation (6.8). The tracking variable is a very important part of the control for the HPCVD problem, since there is some delay between the introduction of controlled species at the inlet and the arrival of that species at the substrate surface (where the tracking signal d^M as well as the absorption measurement $z^M(x^M)$ are evaluated). The tracking variable anticipates this delay, whereas the direct state feedback term $-R^{-1}\mathcal{B}^T\Pi(x^M)x^M$ cannot do so.

The application of the above feedback control to the HPCVD system must also include a state estimator based on the measurement discussed in Section 6.3. The control described in equation (6.8) is given in terms of the state variables x^M (the gas-phase POD coefficients and ROSK model species concentrations), but these are not known. We have only a partial measurement, the optical absorption across the width of the reactor, with which to find an estimate x^M_e of the actual state. There is no direct measurement of the surface state variables used in this model at present. Rewritten in

terms of the POD basis representation, the absorption measurement in equation (6.4) becomes

$$\begin{aligned} z^M(t) &= 1 - \exp\left(-\frac{4\pi W_0}{\lambda} \sum_{n=1}^{3} \sum_{i=1}^{M_n} \frac{b_n}{W_n} \left[\int_{\mathcal{W}} \Phi_{ni}(\overrightarrow{x}) d\overrightarrow{x} \right] y_{ni}(t) \right) \\ &= 1 - \exp\left(-C_0^M x^M(t) \right) \\ &= c(x^M(t)). \end{aligned} \tag{6.15}$$

This measurement is nonlinear in the state, as are the dynamics in the system (6.5), so we will use the nonlinear techniques for state estimation from Section 5.4.

The estimated state will be represented by an ordinary differential equation similar to the state equation, with a gain matrix found using a state-dependent Riccati equation. With the feedback control (6.8) given in terms of the estimated state, the actual and estimated states are coupled in the system given by

$$\left\{ \begin{array}{rcl} \dot{x}^M &=& \mathcal{A}(x^M)x^M - \mathcal{B}R^{-1}\mathcal{B}^T \left[\Pi(x_e^M)x_e^M + s(t, x_{nom}^M) \right] \\ \dot{x}_e^M &=& \mathcal{A}(x_e^M)x_e^M - \mathcal{B}R^{-1}\mathcal{B}^T \left[\Pi(x_e^M)x_e^M + s(t, x_{nom}^M) \right] \\ & & + L^M(x_e^M) \left[z^M - c(x_e^M) \right], \end{array} \right. \tag{6.16}$$

where the state estimation gain is found by

$$L^M(x_e^M) = \Sigma(x_e^M)(C_0^M)^T V^{-1} \tag{6.17}$$

from the solution $\Sigma(x_e^M)$ to the dual state-dependent Riccati equation

$$\Sigma(x_e^M)\mathcal{A}^T(x_e^M) + \mathcal{A}(x_e^M)\Sigma(x_e^M) - \Sigma(x_e^M)(C_0^M)^T V^{-1} C_0^M \Sigma(x_e^M) + U = 0. \tag{6.18}$$

We set the weights in the estimation problem as $U = I_M$ and $V = r_3$, with r_3 a third design parameter we can choose in order to vary the emphasis in the control/estimator formulation. As was mentioned in Section 5.4, for the purposes of finding the estimator gain in equations (6.17)-(6.18) the nonlinear measurement function $z^M = c(x^M)$ is linearized about the origin, resulting in

$$\begin{aligned} z_0^M(t) &= c(0) + \frac{\partial c}{\partial x^M}(0) x^M(t) \\ &= C_0^M x^M(t). \end{aligned}$$

The nonlinearity of the measurement function does remain in the estimator system (6.16) itself. The estimator gain SDRE (6.18) will be solved using the power series approximation in an analogous fashion to equations (6.9)-(6.13), splitting $\mathcal{A}(x^M)$ into constant and state-dependent parts as before and using $N_P = 5$ power series terms.

6.5 Results and Analysis

The POD modes for each gas-phase species were found in the manner described in Section 3.3, from sets of 100 snapshots for each species (obtained by a FIDAP finite element simulation of the species transport process). The percentage of data variability contained in the first k POD modes is plotted in Figure 6.3, for each of the three species. This shows that the original snapshot data is very well

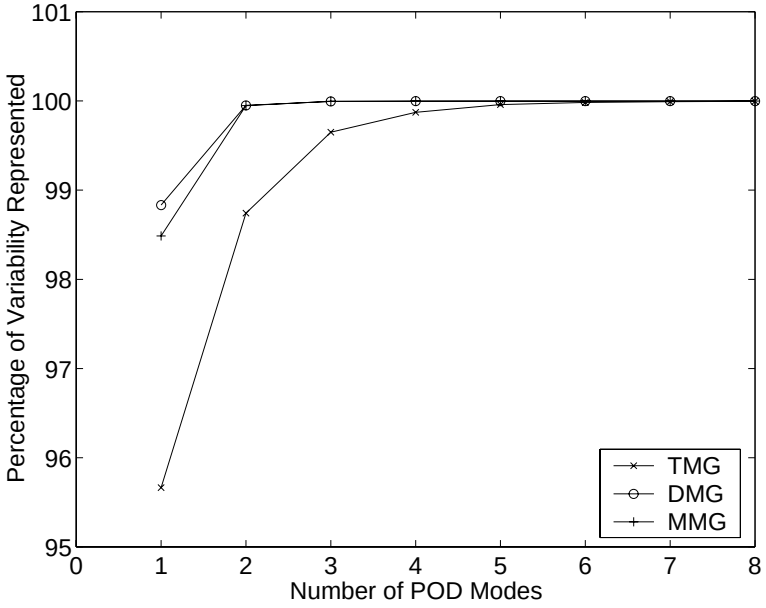

Figure 6.3: Data variability contained in the first few POD modes, for each species.

represented by 6 modes or fewer for the TMG mass fraction, and 3 modes or fewer for the other two species, suggesting the use of no more than 12 POD modes in the reduced order model. This is in contrast with the 3×47131 basis functions that the generic finite element procedure used in the reactor simulation which found the POD snapshots.

However, while variability may be a good guide to the accuracy of the reduced order model as a representation of the full system, the effectiveness of the reduced order model as the basis of feedback control is another issue. For this purpose a study of the controllability and observability properties of the reduced order system may be helpful. The controllability matrix $\mathcal{C}$ for the linear reduced order system in equation (6.1) is given by

$$\mathcal{C}(A^{M_G}, B^{M_G}) = \left[B^{M_G} \mid A^{M_G}B^{M_G} \mid (A^{M_G})^2 B^{M_G} \mid ... \mid (A^{M_G})^{M_G-1} B^{M_G}\right].$$

Using standard results from control theory [109], the M_G-dimensional system (6.1) is controllable if and only if the rank of $\mathcal{C}$ is equal to M_G, and in general the larger the difference between the controllability rank and the system rank, the more difficult the control of the system will be. In a similar manner the observability matrix is given by

$$\mathcal{O}(A^{M_G}, C_0^{M_G}) = \left[C_0^{M_G} \mid (A^{M_G})^T C_0^{M_G} \mid ((A^{M_G})^2)^T C_0^{M_G} \mid ... \mid ((A^{M_G})^{M_G-1})^T C_0^{M_G}\right]$$

for the system (6.1) with the linearized measurement given in terms of the gas-phase states by

$$z_0^{M_G}(y^{M_G}) = C_0^{M_G} y^{M_G}.$$

The system is considered observable if the rank of $\mathcal{O}$ is equal to M_G, and an increase in the difference between the two generally indicates a greater difficulty in observing the system state.

In other studies the ranks of the controllability and observability matrices have sometimes been useful in choosing the number of modes to use in the control design. For example, the reduced order control in [110, 111] was found to be less effective when more modes were added beyond a certain point, and it was suggested that the increase in the reduced order system rank while the controllability rank remained the same may have been the explanation. In contrast, in [7] a smaller POD system (in theory controllable, as the system rank matched the controllability rank of 10) was found to produce a significantly less effective control than a larger POD system (in which the controllability rank was also 10 but the system rank was 19). Also, it should be noted that all these ranks are for the reduced order system, but the eventual goal for any model of this type is to apply the reduced order control to the full high-order system, making it questionable whether controllability actually holds in the full problem even if true for the reduced order system. Thus the controllability rank appears to be something which should be considered, but sticking strictly to it will not always produce optimal results.

The ranks of the controllability and observability matrices increase as the number of POD modes increases, up to a certain point. In the HPCVD reduced order model the largest the controllability rank becomes is 5, and the largest the observability rank becomes is 6. In order to remain very close to the controllability and observability ranks in the POD system, we will use $M_G = 6$ modes total (2 for TMG, 2 for DMG, and 2 for MMG) in constructing the feedback control and estimator. One direction of future work on this problem would be to try a larger system with $M_G = 10$ or 12 modes total. Although by the theory above this would likely be less controllable and observable, it would be interesting to compare the results of a larger system with the 6-mode system and see which is more effective for this problem.

In implementing this problem numerically, the system (3.9) is nondimensionalized before the Galerkin procedure is used to convert it into an ODE system, using reference values of $L_0 = 10^{-3}$ m, $D_0 = 10^{-5}$ m^2/s, and $\rho_0 = 3.04$ kg/m^3, and extrapolating the other scaling values from them. The surface model is nondimensionalized as well before the simulations are run. From the FIDAP-generated snapshots, the POD modes and the coefficient matrices A^{M_G}, B^{M_G}, $H_q^{M_G}$ and $C_0^{M_G}$ are calculated using C programs. The matrix calculations involved in setting up the control formulation are done in MATLAB programs, including the determination of Riccati equation solutions with "*are*" or "*lqr*", and the solution of the TPBV problem for the tracking variable using codes written by the author. The dynamical equations in the actual control problem are solved in MATLAB as well using the "*ode23s*" function. The initial estimated state $x_e^M(0) = O(10^{-8})$ is slightly off of the actual state initial condition $x^M(0) = 0$, which represents a reactor empty of all but the carrier gas, and no species on the surface except the non-state-variable activated phosphorus. The slightly inaccurate estimated state is included to analyze the effectiveness of the state estimation method at recovering to accurately represent the actual state.

We will look at the feedback control of the reduced order system using $M = 6$ POD modes, tracking a thickness growth profile shaped as in Figure 6.2. A series of simulations was done with the values of the design parameters $r_2 = 0$ and $r_3 = 4000$ kept constant, while trying different values for the third parameter, $r_1 = 10^{-9}$, 10^{-8}, 10^{-7}. The resulting film thicknesses from the controls using these three parameter values are plotted in Figure 6.4, tracking the growth up to one monolayer of GaP (with dimensionless time units equivalent to 0.1 s). This plot shows very good tracking of the desired thickness profile, with the thickness moving closer to the desired curve as the value of r_1 for the thickness weighting term in the cost functional increases, although even the $r_1 = 10^{-9}$ output is very close to the target profile. The control inputs u which were used to achieve these three controlled trajectories are plotted in Figure 6.5, with the (scaled down) desired trajectory there as well to show where in the target trajectory particular parts of the control behavior match up. Each control here is a roughly pyramid-shaped, or hat-shaped, pulse. Each goes slightly negative at the beginning and end of the hat shape, and there is a very sharp spike in each located immediately at the start of the time interval considered, before the hat shape itself starts. As r_1 increases, both the hat peak and the negative portions of the control become larger. The larger r_1 values (and so larger Q_1 and relatively smaller R in the cost functional) allow greater extremes in the control, which then force the thickness ramp to have sharper corners and more closely approach the rounded-ramp-shaped desired growth profile.

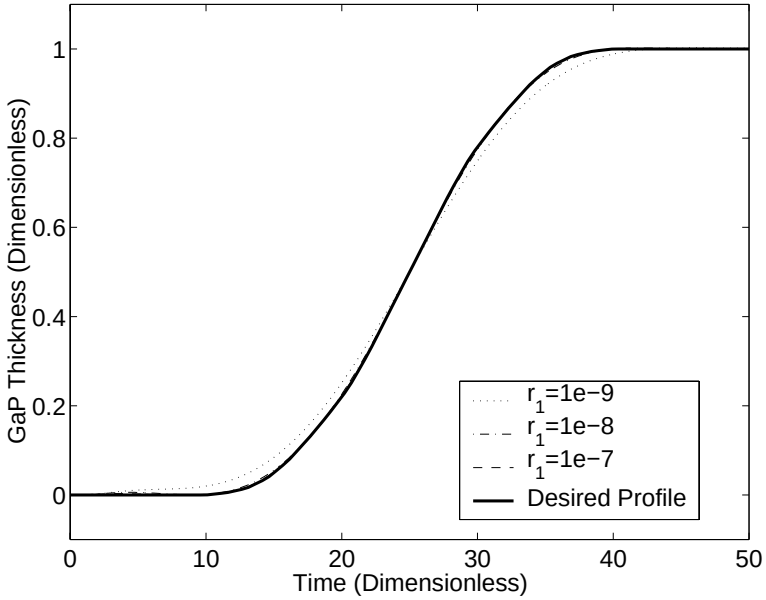

Figure 6.4: Controlled thickness profiles for various r_1 values.

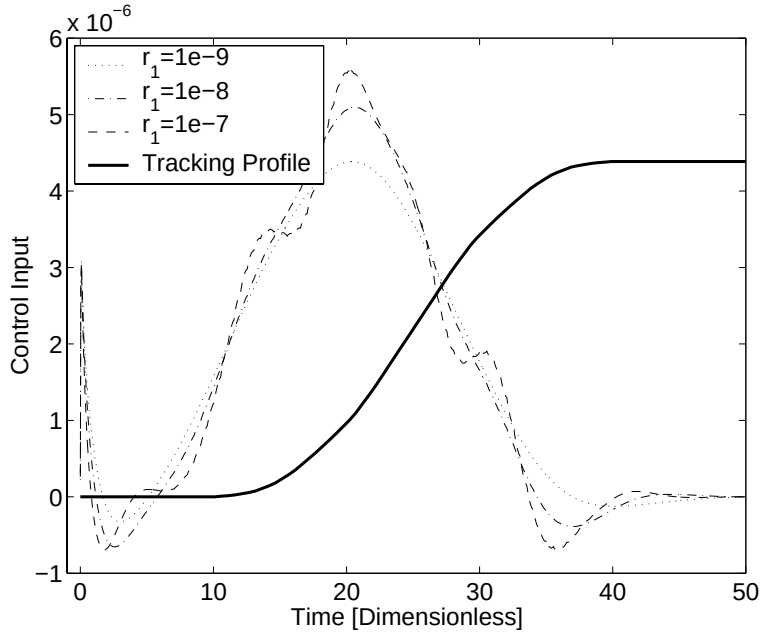

Figure 6.5: Control inputs for various r_1 values, with not-to-scale tracking profile shown.

The initial spike in the control input appears to be related to the slightly inaccurate initial estimated state which we used, although it may also be a very brief period which the control/estimator needs to react to the behavior of the system. Figure 6.6 shows the root-mean-square error between the nodal point values of the gas-phase mass fractions obtained from the actual and estimated states,

$$\text{RMS} = \sqrt{\frac{1}{3N}\left[\overrightarrow{Y}^{N}(t) - \overrightarrow{Y}_{e}^{N}(t)\right]^{2}},$$

during the controlled simulations. Note that this decreases extremely rapidly (the time interval

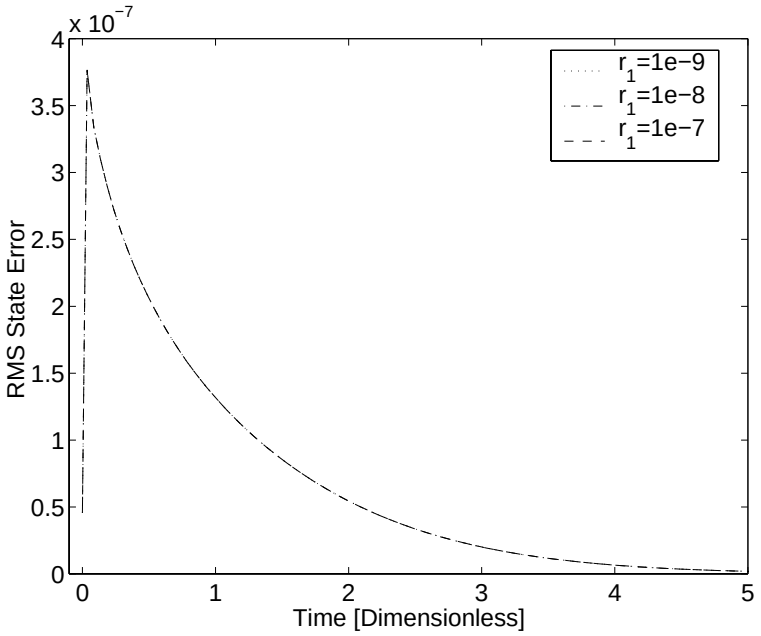

Figure 6.6: State estimation error amounts for various r_1 values.

in this figure is different from that in the thickness and control figures), so that it is almost zero by the nondimensional time 5. This error curve is the same for all three values of r_1, and shows that the estimator is performing well at approximating the actual state from the nonlinear partial measurement. It also implies that the very sudden, very short control spike may be coming in to close the gap between the estimate and the actual state. Other simulations were run with the value of the design parameter $r_3 = 1$. This much smaller value changes the weights in the state estimation cost functional so that more "control" is allowed to close the difference between the estimated and actual states more quickly. This results in the initial control spike being much larger, going from

-11 to 7, in comparison with the spike in Figure 6.5, which has a limited range of 0 to 3. Increasing r_3 much beyond 4000 results in the system becoming badly scaled and "*ode23s*" being unable to solve it.

Without constraints on the possible values the control input can take, it is possible for the control to go negative, as can be seen in Figure 6.5. Since the physical meaning of the control input in the HPCVD problem is the TMG mass fraction at the inlet, having negative values of the control is undesirable, and would be impossible to realize in the physical reactor. To remove this nonphysical behavior in the related problem in the previous study [7], the control was "clipped", or set to 0 whenever the control formulation gave a negative value. In those gas-phase-only simulations, the feedback control successfully tracked the surface flux profile, even with this restriction. In the control problems in this chapter we attempted to use the same "clipping" technique to eliminate negative control inputs, but found that in this case the clipped control could not make the problem converge. There are a few aspects of the HPCVD problem which can help to explain this ineffectiveness.

One important thing to consider is the target profile. The target thickness profile is a upward ramp with rounded corners. If the corners were not rounded at all, it would be virtually impossible to track closely, since sharp corners are extremely difficult to control, especially in this case. Any sharp jump in the gallium input at the inlet (such as a block-shaped pulse) is going to be smoothed out by diffusion in the reactor before it arrives at the surface. In the surface reaction layer, even with a fairly quick reaction rate for the formation of GaP, there is still some time needed for the reaction to happen (likewise for the activation of the surface gallium); since the rate of formation is proportional to the amount of active gallium, as it gets used up the reaction will go slower, prolonging and slowing the end of the film growth process. The way the feedback control adjusts to this situation is by sending negative gallium concentrations out to eliminate the gallium remaining on the surface at the end of the desired growth period. This difficulty is naturally lessened when the corners of the desired thickness ramp function are smoothed out. With the profile shown in Figures 6.4 and 6.5 there is only minimal negative control, especially with the parameter $r_1 = 10^{-9}$.

A different set of simulations using a more ambitious, less rounded target profile was done as well. In this profile the ramp function is less smoothed out, with a growth period of 20 time units compared to 30 in the previous simulations. All the design parameters are the same as above except r_3, for which the value which gave the best results was $r_3 = 1000$, and the time constants $T_p = \Delta = 30$. The thickness profile results for these simulations using the same three values of r_1 as before are plotted in Figure 6.7, with the corresponding control inputs plotted in Figure 6.8,

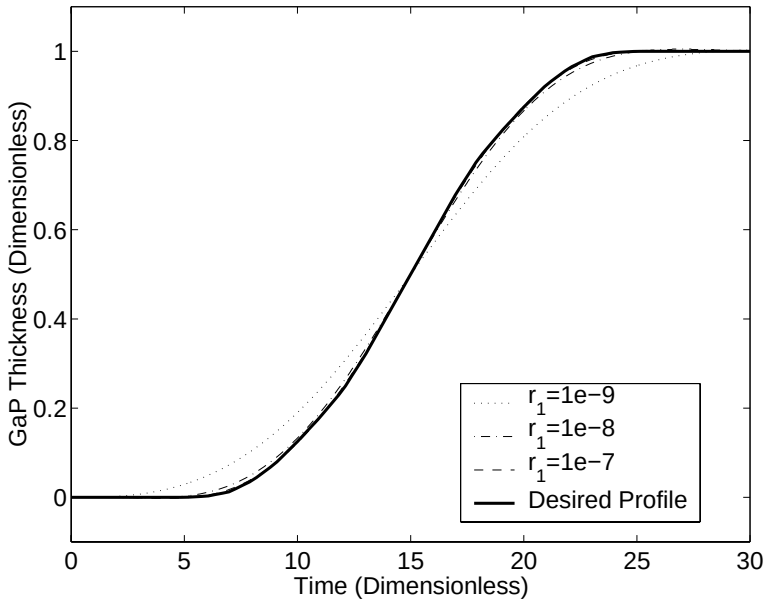

Figure 6.7: Controlled thickness profiles with sharper target profile.

along with the not-to-scale target thickness trajectory. We can track this sharper thickness profile almost as well as the earlier profile, but with greater difficulty in keeping the control nonnegative. Notice that in Figure 6.8 the $r_1 = 10^{-7}$ control input goes down to almost -2 as the growth is levelling off, while in Figure 6.5 it only reaches about $-2/3$. Even the $r_1 = 10^{-9}$ control input goes down to nearly -1 in the initial spike in Figure 6.8, while only going to about $-1/3$ near the start in Figure 6.5. It might be possible to remove the negative control inputs entirely with enough rounding and slowing of the desired film growth. However, this would limit the rate at which the film could be grown. To determine the proper balance between the speed of growth and the ability to control the process, so that we can find what target profile is most desirable yet can be realistically achieved, we will need more information about the HPCVD reactor and model (accurate values of the parameters k_1 and k_{GaP}, etc). The estimator error resulting from the sharper-profile simulations is shown in Figure 6.9. Note that the error here is reduced to very near zero by about time 1.5, significantly earlier than the time 5 which the error in Figure 6.6 needs, as a result of the smaller parameter $r_3 = 1000$ compared to the Figure 6.6 value of $r_3 = 4000$.

Another property of the film thickness tracking problem which contributes to the failure of clipping is that the tracking signal (film thickness) is a cumulative measure of the amount of GaP

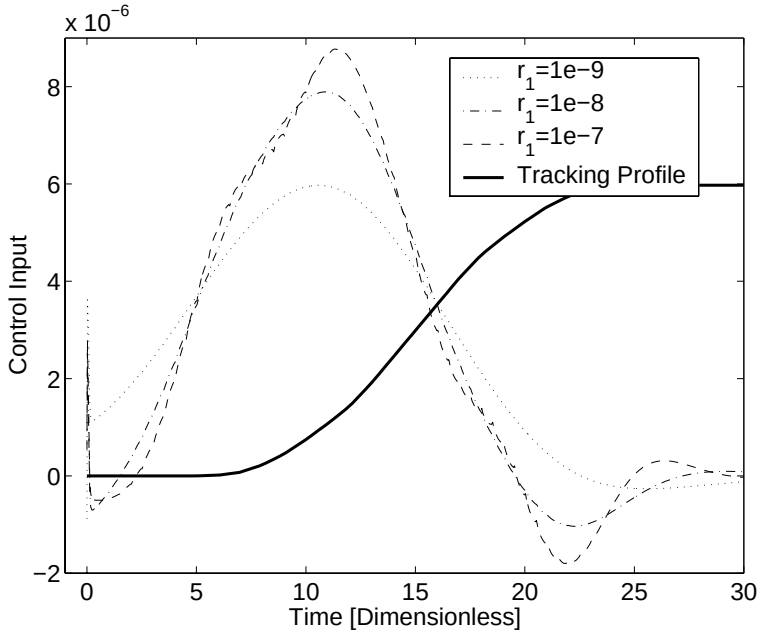

Figure 6.8: Control inputs with sharper target profile (not-to-scale tracking profile shown).

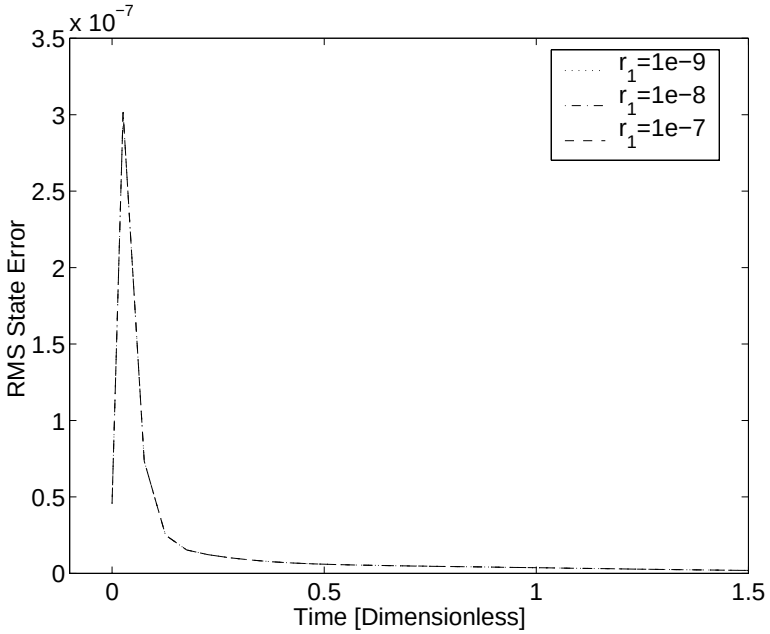

Figure 6.9: State estimation error amounts with sharper target profile.

which has been grown during the cycle up to the current time. In [7], where clipped control was successfully implemented, the tracking signal was the flux of gallium to the surface, which is an instantaneous property of the system. In that case, if a negative control input was clipped near the end of the desired block-shaped flux profile, it would simply lead to a slight trailing over of the gallium flux past the desired stopping time, but no long-term unwanted effects. In fact the target profile in [7] includes a slight trailing end in anticipation of this effect. In contrast, if no negative control is allowed in our current problem, any excess active gallium on the surface which the negative control would have removed instead goes into forming the GaP film, thereby overshooting the upper plateau of the target ramp function. The film thickness will be off of the desired value for the rest of the time period instead of just a short interval, thus adding tremendously to the cost functional. This is the reason for the failure of the clipped feedback control. If the HPCVD model was extended to consider long-term film growth this would be less of a problem, since any overshoot could be compensated for with less growth in the next source vapor pulse cycle. The next phosphorus source pulse could be used to control the leftover active surface gallium through reactions in the next growth period.

It can be noted that in all of the above simulations we have used the value $r_2 = 0$ for the added state weight term $(\overline{d}^M)^T Q_2 \overline{d}^M$ in the cost functional. Some simulations were run with a value of $r_2 = 10^{-9}$, but using this did not seem to be beneficial overall. There did not appear to be any significant undesirable behavior in the state variables which needed to be limited by this term, and diverting the feedback control influence to the other states as well reduced the ability of the control to track the desired thickness profile. Figure 6.10 contains a plot of the thickness trajectory for a simulation with conditions identical to those for Figures 6.7-6.9 except for $r_2 = 10^{-9}$, using the thickness weighting parameter $r_1 = 10^{-9}$. The thickness trajectory using $r_1 = 10^{-9}$ from Figure 6.7 is plotted as well in Figure 6.10, for a comparison of the results using different r_2 values. This shows that the $r_2 = 10^{-9}$ profile is significantly less effective at tracking the desired thickness. The control input behavior changes somewhat with the change in r_2, but the general shape is the same, and the initial spike is even more negative for $r_2 = 10^{-9}$ than for $r_2 = 0$, as can be seen in Figure 6.11.

6.6 Conclusions

In this chapter we have found a reduced order model to represent the species transport in the three-dimensional HPCVD reactor, using 6 POD modes. This was coupled with the modified ROSK

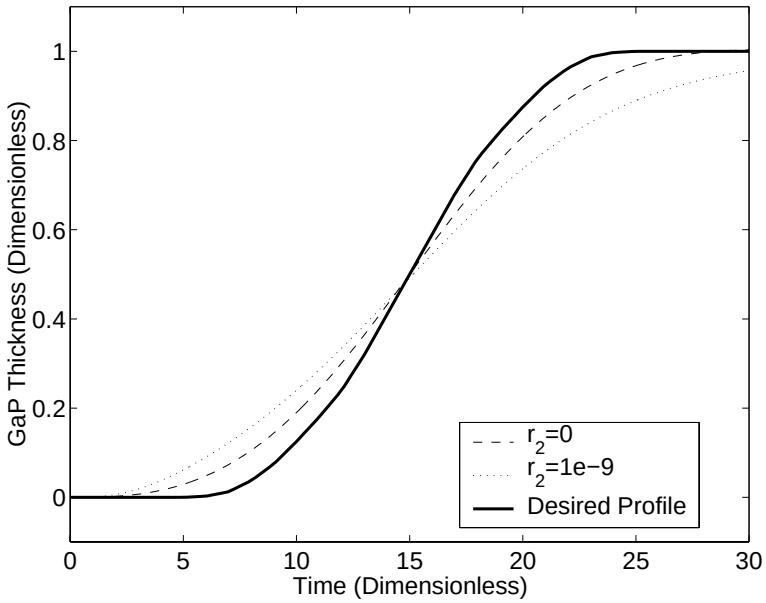

Figure 6.10: Controlled thickness profiles for two r_2 values.

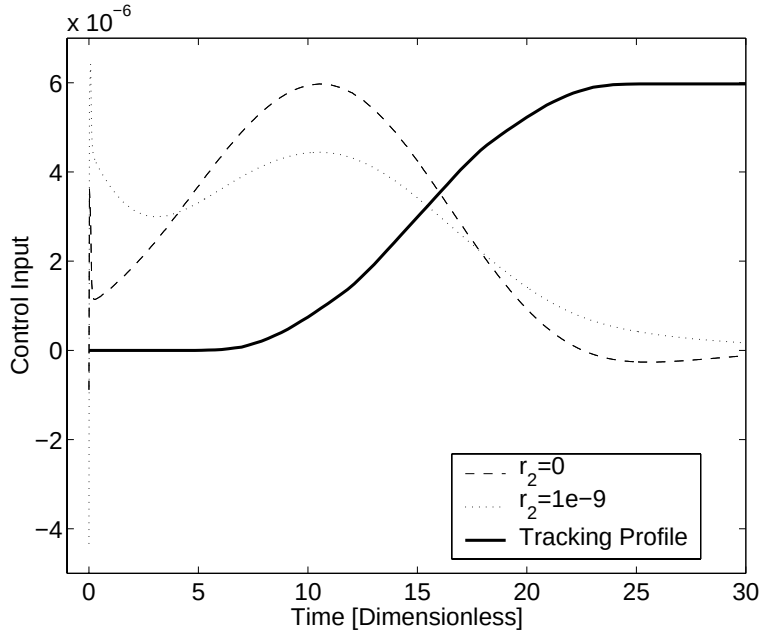

Figure 6.11: Control inputs for two r_2 values (not-to-scale tracking profile shown).

model of the surface reactions to complete the reduced order film growth model. A feedback control tracking the GaP film thickness was constructed for this system, using a nonlinear partial measurement for estimation of the state. The nonlinear control was effective at regulating the system and tracking the film growth on the substrate, and the nonlinear estimator using the optical absorption measurement approximated the actual state very well, establishing the effectiveness of these methods on a larger nonlinear system than the simple test problems they were applied to previously. One area of concern is the presence of negative control inputs in the problem, which are not physically possible in the actual HPCVD reactor. The method of "clipping" these values to zero failed to produce a usable control, leaving this as a problem to be dealt with in the future, either by changing the target thickness profile again, by considering the more long-term growth control problem, or with some other means. Application of the reduced order control developed in this chapter to the finite element representation of the gas-phase system might give insight into how well the POD system describes the physical species transport dynamics, as well as a prediction of how well the reduced order control might be at controlling the physical reactor.

Chapter 7

Summary and Future Research Directions

In the work described in the previous chapters we have developed a mathematical model of the thin film growth process in a high-pressure chemical vapor deposition reactor. This model combines the gas-phase flow of the source species to the substrate with the growth of the film on that substrate surface. The film growth itself is represented by a reduced order surface kinetics model, with its structure validated and parameters determined through a series of low-pressure CVD experiments. The previously established quasi-steady gas flow and species transport model, including gas-phase reactions, is extended here to three dimensions, and a realistic technique for partial measurement of the process is added. The gas-phase model is transformed into a reduced order system using the proper orthogonal decomposition method to find a basis tailored to this particular problem. The two parts of the HPCVD model are linked through the gas-phase species flux to the substrate surface.

Since we have low order models representing each part of the film deposition process, we can run the model simulations very fast. This enables us to construct a control problem using this model to generate a real-time feedback control of the film growth process. Due to the lack of a single established method for feedback control of nonlinear systems, we performed a comparison study of five methods available in the literature, evaluating their performance on some simple test problems with similar properties to the HPCVD model. Choosing the state-dependent Riccati equation method, which was effective on nearly all of the test examples, we extended it from a basic stabilizing control to a control tracking a certain desired signal. We also extended state estimation techniques from the literature into a procedure which more fully incorporates the nonlinear nature of the problem, using the SDRE in this as well. These new tracking control and state estimation methods were also tested on simple problems to compare their effectiveness with that of the standard

linear methods.

We then applied the nonlinear feedback control techniques which we developed to the combined reduced order gas-phase and surface kinetics models, running simulations in which the inlet TMG mass fraction was used to control the model and track a desired film thickness growth profile. A representation of the nonlinear optical absorption measurement available in the physical reactor was used for state estimation. The resulting feedback control was successful at estimating the actual state and tracking the desired thickness trajectory, with the drawback of some slightly negative control input values near the film growth starting and stopping times. The importance of choosing the right target film growth pattern was discovered; trying to force the growth to be too fast, or to start and stop too sharply, resulted in large negative control input portions. On the other hand, making the target growth profile too slow will result in the loss of one of the advantages of the HPCVD process, its speed in comparison with low-pressure methods.

There are several directions in which this research work could be expanded on in the future. In Chapter 6 we have derived a reduced order model-based control and implemented it on the reduced order model of the HPCVD problem. However, the success of this is not necessarily indicative of a success upon application of the reduced order control to the full system. This can be tested by implementing the model on the high-order finite element representation in the following manner:

$$\begin{cases} \dot{x}^{N_C} = \mathcal{A}^{N_C}(x^{N_C})x^{N_C} - \mathcal{B}^{N_C}R^{-1}\mathcal{B}^T\left[\Pi(x_e^M)x_e^M + s(t, x_{nom}^M)\right] \\ \dot{x}_e^M = \mathcal{A}(x_e^M)x_e^M - \mathcal{B}R^{-1}\mathcal{B}^T\left[\Pi(x_e^M)x_e^M + s(t, x_{nom}^M)\right] + L^M(x_e^M)\left[z^{N_C}(x^{N_C}) - z^M(x_e^M)\right], \end{cases}$$

where $N_C = 3N + M_S$ is the size of the combined full finite element gas-phase model and surface model, and $\mathcal{A}^{N_C}(x^{N_C})$, $\mathcal{B}^{N_C}$ and $z^{N_C}(x^{N_C})$ are the versions of the model components in equations (6.5) and (6.15) incorporating the full finite element representation of the gas-phase species mass fractions. This would use the optical absorption measurement and track the film thickness as was done in Chapter 6, except now with the actual state in terms of the finite element coefficients instead of the POD coefficients. Beyond this, the ultimate goal is to apply the reduced order model-based feedback control to the physical reactor, which has just been constructed in the Materials Science Department at North Carolina State University.

In order to make the model accurate enough to effectively apply the control based on it to the physical reactor, some refinements will have to be made. Several parameters we have used here are very rough estimates, used to prove the effectiveness of the control on a problem with behavior very similar to that of the physical problem. For the actual reactor, however, we will need to find more accurate values for the ROSK model rate constants and the optical responses for the gas-phase

species, through calibration and experiments in the reactor. In addition, there will be a second measurement technique added in the actual reactor, one related to the PRS measurement described in Chapter 2. The difference is that it originates from behind the substrate, as shown in Figure 6.1, since the high-pressure nature of the reactor interior makes the regular PRS measurement from above the substrate unreliable. The equations modeling this will need to be added to the state estimation portion of the control in order to use this measurement in controlling the physical reactor.

A different avenue of investigation is the POD reduced order model; we have used 6 POD modes in representing the three-dimensional species transport dynamics, but this may not be enough to portray the actual reactor as accurately as we want. Larger models using perhaps 10 or 12 modes could be tried to see whether control authority would improve or not. Application of these different reduced order models to the finite element model would be especially useful in determining which would be best for controlling the physical reactor. The nonlinear control methodologies developed here could also be studied further. We have primarily investigated their effectiveness when applied to certain test problems, and later the HPCVD problem. It would be useful to have an analytical look at the properties of these methods, such as robustness and local and global stability for the control as well as the estimator, so we can determine under what conditions these can be achieved.

Another important point is that in this work we have looked only at a single cycle of the deposition process, concentrating on the small-time-scale behavior of the system. While this is important, controlling the long term growth behavior (such as that shown in the PRS measurements in Figure 2.4) over a period of hundreds of cycles is another important goal. This will mean extending the model to include the gas-phase phosphine transport and decomposition in addition to that of TMG, and changing the ROSK model again to include the surface phosphorus dynamics. Some modifications to the control problem formulation may prove necessary in this event as well. A benefit of extending the control problem into the long term would be that the problem with "clipping" negative controls might be removed, since overshooting a target film thickness could be compensated for in the next source vapor pulse cycle. The addition of the pseudo-PRS measurement would also be helpful in controlling the long term behavior and removing the clipping problem, since the optical absorption measurement focuses only on the gas phase while the pseudo-PRS measurement provides a more direct observation of the cumulative film thickness growth. A further possibility is that the problem could be extended to look at the growth of films such as $Ga_{1-x}In_xP$, where both the film thickness and film composition can be controlled, by adding more species and reactions to both the gas-phase and surface models.

List of References

[1] Gevelber, M., Toledo-Quiñones, M., and Bufano, M., "Towards Closed-Loop Control of CVD Coating Microstructures," Materials Science and Engineering A, Vol. 209, pp. 377-383 (1996).

[2] Zhou, J. J., Li, Y., Pacheco, D., Lee, H. P., and Liu, X., "Virtual Control Simulator for Closed-Loop Epitaxial Growth," Journal of Crystal Growth, Vol. 175, pp. 52-60 (1997).

[3] Warnick, S. C., and Dahleh, M. A., "Feedback Control of MOCVD Growth of Submicron Compound Semiconductor Films," IEEE Transactions on Control Systems Technology, Vol. 6, pp. 62-71 (1998).

[4] Dietz, N., Woods, V., Ito, K., and Lauko, I., "Real-Time Optical Control of $Ga_{1-x}In_xP$ Film Growth by p-Polarized Reflectance," Journal of Vacuum Science and Technology A, Vol. 17, pp. 1300-1306 (1999).

[5] Ly, H. V., and Tran, H. T., "Proper Orthogonal Decomposition for Flow Calculations and Optimal Control in a Horizontal CVD Reactor," CRSC Technical Report CRSC-TR98-13, N.C. State University (1998), and Quarterly of Applied Mathematics, to appear.

[6] Kepler, G. M., Tran, H. T., and Banks, H. T., "Reduced Order Model Compensator Control of Species Transport in a CVD Reactor," CRSC Technical Report CRSC-TR99-15, N.C. State University (1999), and Optimal Control Applications and Methods, to appear.

[7] Kepler, G. M., Tran, H. T., and Banks, H. T., "Compensator Control for Chemical Vapor Deposition Film Growth Using Reduced Order Design Models," CRSC Technical Report CRSC-TR99-41, N.C. State University (1999), and IEEE Transactions on Semiconductors, submitted.

[8] Dietz, N. and Ito, K., "Real-Time Optical Characterization of GaP Heterostructures by p-Polarized Reflectance," Thin Solid Films, Vol 313, pp. 614-619 (1998).

[9] Aspnes, D. E. and Dietz, N., "Optical Approaches for Controlling Epitaxial Growth," Applied Surface Science, Vol 132, pp. 367-376 (1998).

[10] Dietz, N. and Bachmann, K. J., "Real-Time Monitoring of Epitaxial Processes by Parallel-Polarized Reflectance Spectroscopy," MRS Bulletin, Vol 20, pp. 49-55 (1995).

[11] Dietz, N. and Bachmann, K. J., "p-Polarized Reflectance Spectroscopy: A Highly Sensitive Real-Time Monitoring Technique to Study Surface Kinetics Under Steady State Epitaxial Deposition Conditions," Vacuum, Vol 47, pp. 133-140 (1996).

[12] Dietz, N., Sukidi, N., Harris, C., and Bachmann, K. J., "Real-Time Monitoring of Surface Processes by p-Polarized Reflectance," Journal of Vacuum Science and Technology A, Vol 15, pp. 807-815 (1997).

[13] Dietz, N., Miller, A., and Bachmann, K. J., "Real-Time Monitoring of Homoepitaxial and Heteroepitaxial Processes by p-Polarized Reflectance Spectroscopy," Journal of Vacuum Science and Technology A, Vol 13, pp. 153-155 (1995).

[14] Dietz, N., Miller, A., Kelliher, J. T., Venables, D., and Bachmann, K. J., "Migration-Enhanced Pulsed Chemical Beam Epitaxy of GaP on Si(001)," Journal of Crystal Growth, Vol 150, pp. 691-695 (1995).

[15] Dietz, N., Rossow, U., Aspnes, D., and Bachmann, K. J., "Real-Time Optical Monitoring of Epitaxial-Growth – Pulsed Chemical Beam Epitaxy of GaP and InP Homoepitaxy and Heteroepitaxy on Si," Journal of Electronic Materials, Vol 24, pp. 1571-1576 (1995).

[16] Bachmann, K. J., Dietz, N., Miller, A. E., Venables, D., and Kelliher, J. T., "Heteroepitaxy of Lattice-Matched Compound Semiconductors on Silicon," Journal of Vacuum Science and Technology A, Vol 13, pp. 696-704 (1995).

[17] Bachmann, K. J., Rossow, U., Sukidi, N., Castleberry, H., and Dietz, N., "Heteroepitaxy of GaP on Si(001)," Journal of Vacuum Science and Technology B, Vol 14, pp. 3019-3029 (1996).

[18] Dietz, N., Rossow, U., Aspnes, D. E., and Bachmann, K. J., "Real-Time Optical Monitoring of Heteroepitaxial Growth Processes on Si Under Pulsed Chemical Beam Epitaxy Conditions," Applied Surface Science, Vol 102, pp. 47-51 (1996).

[19] Dietz, N., Sukidi, N., Harris, C., and Bachmann, K. J., "Real-time Characterization of the Optical Properties of an Ultra-Thin Surface Reaction Layer During Growth", Materials Research Society Symposia Proceedings, Vol. 441, pp. 39-44 (1997).

[20] Bachmann, K. J., Sukidi, N., Hopfner, C., Harris, C., Dietz, N., Tran, H. T., Beeler, S., Ito, K., and Banks, H. T., "Real-Time Monitoring of Steady-State Pulsed Chemical Beam Epitaxy by p-Polarized Reflectance," Journal of Crystal Growth, Vol 183, pp. 323-337 (1998).

[21] Dietz, N., Sukidi, N., Harris, C., and Bachmann, K. J., in Conference Proceedings of IPRM-9 1997, ISSN 1092-8669, p. 521 (1997).

[22] Bachmann, K. J., Hopfner, C., Sukidi, N., Miller, A. E., Harris, C., Aspnes, D. E., Dietz, N. A., Tran, H. T., Beeler, S., Ito, K., Banks, H. T., and Rossow, U., "Molecular Layer Epitaxy by Real-Time Optical Process Monitoring," Applied Surface Science, Vol 112, pp. 38-47 (1997).

[23] Heavens, O. S., *Optical Properties of Thin Solid Films*, London: Butterworths, 1955.

[24] Ward, L., *The Optical Constants of Bulk Materials and Films*, 2nd ed., London: IOP, 1994.

[25] Li, S. H., Larsen, C. A., Buchan, N. I., Stringfellow, G. B., Kosar, W. P., and Brown, D. W., "Study of Tertiarybutylphosphine Pyrolysis Using a Deuterated Source," Journal of Applied Physics, Vol 65, pp. 5161-5165 (1989).

[26] Fan, G. H., Hoare, R. D., Pemble, M. E., Povey, I. M., and Taylor, A. G., "Gas-Phase Monitoring of Reactions Under InP MOVPE Growth-Conditions for the Decomposition of Tertiarybutyl Phosphine and Related Precursors," Journal of Crystal Growth, Vol 124, pp. 49-55 (1992).

[27] Murrell, A. J., Wee, A. T. S., Fairbrother, D. H., Singh, N. K., Foord, J. S., Davies, G. J., and Andrews, D. A., "Surface Studies of the Thermal-Decomposition of Triethylgallium on GaAs(100)," Journal of Crystal Growth, Vol 105, pp. 199-202 (1990).

[28] Burns, G., *Solid State Physics*, p. 461, Orlando, Florida: Academic, 1985.

[29] Kelley, C. T., *Iterative Methods for Optimization*, SIAM, 1999.

[30] Kelley, C. T., "Detection and Remediation of Stagnation in the Nelder-Mead Algorithm Using a Sufficient Decrease Condition," SIAM Journal on Optimization, Vol 10, pp. 43-55 (1999).

[31] Bortz, D. M. and Kelley, C. T., "The Simplex Gradient and Noisy Optimization Problems," in *Computational Methods for Optimal Design and Control*, edited by Borggaard, J. T., Burns, J., Cliff, E., and Schreck, S., pp. 77-90, Boston, Massachusetts: Birkhauser, 1998.

[32] Bird, R. B., Stewart, W. E., and Lightfoot, E. N., *Transport Phenomena*, New York, New York: John Wiley and Sons, 1960.

[33] Slattery, J. C., *Momentum, Energy, and Mass Transfer in Continua*, New York, New York: McGraw-Hill, 1972.

[34] *FIDAP (Fluid Dynamics Analysis Package) 7.0 Theory Manual*, Evanston, Illinois: Fluid Dynamics International, 1993.

[35] Shadid, J. N., Moffat, H. K., Hutchinson, S. A., Hennigan, G. L., Devine, K. D., and Salinger, A. G., *MPSalsa: A Finite Element Computer Program for Reacting Flow Problems. Part 1 – Theoretical Development* (Sandia National Laboratories Report), Springfield, Virginia: National Technical Information Service, 1996.

[36] Svehla, R. A., NASA Technical Report R-132, 1962.

[37] Bayazitoglu, Y., and Ozisik, M. N., *Elements of Mass Heat Transfer*, New York, New York: McGraw Hill, 1988.

[38] Lide, D. R., and Kehiaian, H. V., *CRC Handbook of Thermophysical and Thermochemical Data*, Boca Raton, Florida: CRC Press, 1994.

[39] Shah, R. K., and London, A. L., *Laminar Flow Forced Convection in Ducts*, New York, New York: Academic Press, 1978.

[40] Buchan, N. I., and Jasinski, J. M., "Calculation of Unimolecular Rate Constants for Common Metalorganic Vapor-Phase Epitaxy Precursors via RRKM Theory," Journal of Crystal Growth, Vol 106, pp. 227-238 (1990).

[41] Larsen, C. A., Buchan, N. I., Li, S. H., and Stringfellow, G. B., "Decomposition Mechanisms of Trimethylgallium," Journal of Crystal Growth, Vol 102, pp. 103-116 (1990).

[42] Hirschfelder, J. P., Curtis, C. F., and Bird, R. B., *Molecular Theory of Gases and Liquids*, New York, New York: Wiley and Sons, 1954.

[43] Babuška, I., "The Finite Element Method With Penalty," Mathematics of Computation, Vol 27, pp. 221-228 (1973).

[44] Barrett, J. W., and Elliot, C. M., "Finite Element Approximation of the Dirichlet Problem Using the Boundary Penalty Method," Numerische Mathematik, Vol 49, pp. 343-366 (1986).

[45] Karhunen, K., "Zur Spektral Theorie Stochasticher Prozesse," Annales Academiae Scientiarum Fennicae, Series A1, Mathematica Physica, Vol 37 (1946).

[46] Loève, M., "Functions Aleatoire de Second Ordre," Compte Rend. Acad. Sci. (Paris), Vol 220 (1945).

[47] Jackson, J. E., *A User's Guide to Principal Components*, New York, New York: John Wiley and Sons, 1991.

[48] Fukunaga, K., *Introduction to Statistical Pattern Recognition*, New York, New York: Academic Press, 1972.

[49] Lumley, J. L., *Stochastic Tools in Turbulence*, New York, New York: Academic Press, 1970.

[50] Berkooz, G., *Turbulence, Coherent Structures, and Low-Dimensional Models*, Ph.D. dissertation, Cornell University, 1991.

[51] Berkooz, G., "Observations on the Proper Orthogonal Decomposition," in Gatski, T. B., Sarkar, S., and Speziale, C. G., eds., *Studies in Turbulence*, New York, New York: Springer-Verlag, 1992, pp. 229-247.

[52] Holmes, P. J., Lumley, J. L., Berkooz, G., Mattingly, J. C., and Wittenberg, R. W., "Low-Dimensional Models of Coherent Structures in Turbulence," Physics Reports – Review Section of Physics Letters, Vol 287, pp. 338-384 (1997).

[53] Iollo, A., Lanteri, S., and Désidéri, J. A., "Stability Properties of POD-Galerkin Approximations for the Compressible Navier-Stokes Equations," Theoretical and Computational Fluid Dynamics, Vol 13, pp. 377-396 (2000).

[54] Theodoropoulou, A., Adomaitis, R. A., and Zafiriou, E., "Model Reduction for Optimization of Rapid Thermal Chemical Vapor Deposition Systems," IEEE Transactions on Semiconductor Manufacturing, Vol 11, pp. 85-98 (1998).

[55] Temam, R., *Infinite-Dimensional Dynamical Systems in Mechanics and Physics*, New York, New York: Springer-Verlag, 1988.

[56] Riesz, F., and Szokefalvi-Nagy, B., *Functional Analysis*, New York, New York: Ungar, 1955.

[57] Anderson, B. D. O., and Moore, J. B., *Optimal Control: Linear Quadratic Methods*, Englewood Cliffs, New Jersey: Prentice-Hall, 1990.

[58] Lewis, F. L., and Syrmos, V. L., *Optimal Control*, New York: Wiley, 1995.

[59] Garrard, W. L., "Suboptimal Feedback Control for Nonlinear Systems," Automatica, Vol 8, pp. 219-221 (1972).

[60] Garrard, W. L., and Jordan, J. M., "Design of Nonlinear Automatic Flight Control Systems," Automatica, Vol 13, pp. 497-505 (1977).

[61] Garrard, W. L., Enns, D. F., and Snell, S. A., "Nonlinear Feedback Control of Highly Manoeuvrable Aircraft," International Journal of Control, Vol 56, pp. 799-812 (1992).

[62] Nishikawa, Y., Sannomiya, N., and Itakura, H., "A Method for Suboptimal Design of Nonlinear Feedback Systems," Automatica, Vol 7, pp. 703-712 (1971).

[63] Leake, R. J. and Liu, R. W., "Construction of Suboptimal Control Sequences," SIAM Journal on Control and Optimization, Vol 5, pp. 54-63 (1967).

[64] Saridis, G. N., and Lee, C. S. G., "An Approximation Theory of Optimal Control for Trainable Manipulators," IEEE Transactions on Systems, Man, and Cybernetics, Vol 9, pp. 152-159 (1979).

[65] Beard, R. W., Saridis, G. N., and Wen, J. T., "Galerkin Approximation of the Generalized Hamilton-Jacobi-Bellman Equation," Automatica, Vol 33, pp. 2159-2177 (1997).

[66] Beard, R. W., Saridis, G. N., and Wen, J. T., "Approximate Solutions to the Time-Invariant Hamilton-Jacobi-Bellman Equation," Journal of Optimization Theory and Applications, Vol 96, pp. 589-626 (1998).

[67] Pearson, J. D., "Approximation Methods in Optimal Control," Journal of Electronics and Control, Vol 13, pp. 453-465 (1962).

[68] Burghart, J. A., "A Technique for Suboptimal Control of Nonlinear Systems," IEEE Transactions on Automatic Control, Vol 14, pp. 530-533 (1969).

[69] Wernli, A. and Cook, G., "Suboptimal Control for the Nonlinear Quadratic Regulator Problem," Automatica, Vol 11, pp. 75-84 (1975).

[70] Krikelis, N. J. and Kiriakidis, K. I., "Optimal Feedback Control of Non-linear Systems," International Journal of Systems Science, Vol 23, pp. 2141-2153 (1992).

[71] Cloutier, J. R., D'Souza, C. N., and Mracek, C. P., "Nonlinear Regulation and Nonlinear H_∞ Control Via the State-Dependent Riccati Equation Technique: Part 1. Theory," Proceedings of the First International Conference on Nonlinear Problems in Aviation and Aerospace, Daytona Beach, FL, May 1996.

[72] Halme, A., and Hamalainen, R. P., "On the Nonlinear Regulator Problem," Journal of Optimization Theory and Applications, Vol 16, pp. 255-275 (1975).

[73] Goh, C. J., "On the Nonlinear Optimal Regulator Problem," Automatica, Vol 29, pp. 751-756 (1993).

[74] Ito, K. and Schroeter, J. D., "Reduced Order Feedback Synthesis for Viscous Incompressible Flows," CRSC Technical Report CRSC-TR98-41, N.C. State University (1998).

[75] Kirk, D. E., *Optimal Control Theory*, Englewood Cliffs, New Jersey: Prentice-Hall, 1970.

[76] Sage, A. P., and White, C. C., III, *Optimum Systems Control, 2nd Edition*, Englewood Cliffs, New Jersey: Prentice-Hall, 1977.

[77] Bosarge, W. E., Johnson, O. G., McKnight, R. S., and Timlake, W. P., "The Ritz-Galerkin Procedure for Nonlinear Control Problems," SIAM Journal on Numerical Analysis, Vol 10, pp. 94-110 (1973).

[78] Markman, J., and Katz, I. N., "An Iterative Algorithm for Solving Hamilton-Jacobi Type Equations," SIAM Journal on Scientific Computing, Vol 22, pp. 312-329 (2000).

[79] Markman, J., and Katz, I. N., "Convergence of an Iterative Algorithm for Solving Hamilton Jacobi Type Equations," preprint.

[80] Hofer, E. P., and Tibken, B., "An Iterative Method for the Finite-Time Bilinear Quadratic Control Problem," Journal of Optimization Theory and Applications, Vol 57, pp. 411-427 (1988).

[81] Aganovic, Z., and Gajic, Z., "The Successive Approximation Procedure for Finite-Time Optimal Control of Bilinear Systems," IEEE Transactions on Automatic Control, Vol 29, pp. 1932-1935 (1994).

[82] Cebuhar, W. A. and Costanza, V., "Approximation Procedures for the Optimal Control of Bilinear and Nonlinear Systems," Journal of Optimization Theory and Applications, Vol 43, pp. 615-627 (1984).

[83] Ryan, E. P., "Optimal Feedback Control of Bilinear Systems," Journal of Optimization Theory and Applications, Vol 44, pp. 333-362 (1984).

[84] Tzasfestas, S. G., Anagnostou, K. E., and Pimenides, T. G., "Stabilizing Optimal Control of Bilinear Systems with a Generalized Cost," Optimal Control Applications and Methods, Vol 5, pp. 111-117 (1984).

[85] Werner, R. A. and Cruz, J. B., "Feedback Control Which Preserves Optimality for Systems with Unknown Parameters," IEEE Transactions on Automatic Control, Vol 13, pp. 621-629 (1968).

[86] Qu, Z., "Robust Control of Nonlinear Uncertain Systems without Generalized Matching Conditions," IEEE Transactions on Automatic Control, Vol 40, pp. 1453-1460 (1995).

[87] Bourdache-Siguerdidjane, H. and Fliess, M., "Optimal Control of Non-linear Systems," Automatica, Vol 23, pp. 373-379 (1987).

[88] Banks, H. T., Smith, R. C., and Wang, Y., *Smart Material Structures: Modeling, Estimation and Control*, Chichester, England: Wiley, 1996.

[89] Lasiecka, I. and Triggiani, R., *Differential and Algebraic Riccati Equations with Application to Boundary/Point Control Problems*, New York: Springer-Verlag, 1991.

[90] Cloutier, J. R., D'Souza, C. N., and Mracek, C. P., "Nonlinear Regulation and Nonlinear H_∞ Control Via the State-Dependent Riccati Equation Technique: Part 2. Examples," Proceedings of the First International Conference on Nonlinear Problems in Aviation and Aerospace, Daytona Beach, FL, May 1996.

[91] Thau, F. E., "Observing the State of Non-linear Dynamic Systems," International Journal of Control, Vol 17, pp. 471-479 (1973).

[92] Kou, S. R., Elliott, D. L., and Tarn, T. J., "Exponential Observers for Nonlinear Dynamic Systems," Information and Control, Vol 29, pp. 204-216 (1975).

[93] Mielczarski, W., "Observing the State of a Synchronous Generator – Part 1. Theory," International Journal of Control, Vol 45, pp. 987-1000 (1987).

[94] Hu, X., "On State Observers for Nonlinear Systems," Systems and Control Letters, Vol 17, pp. 465-473 (1991).

[95] Krener, A. J., and Isidori, A., "Linearization by Output Injection and Nonlinear Observers," Systems and Control Letters, Vol 3, pp. 47-52 (1983).

[96] Bestle, D., and Zeitz, M., "Canonical Form Observer Design for Non-linear Time-Variable Systems," International Journal of Control, Vol 38, pp. 419-431 (1983).

[97] Krener, A. J., and Respondek, W., "Nonlinear Observers with Linearizable Error Dynamics," SIAM Journal of Control and Optimization, Vol 23, pp. 197-216 (1985).

[98] Zeitz, M., "The Extended Luenberger Observer for Nonlinear Systems," Systems and Control Letters, Vol 9, pp. 149-156 (1987).

[99] Xia, X., and Gao, W., "Nonlinear Observer Design by Observer Error Linearization," SIAM Journal of Control and Optimization, Vol 27, pp. 199-216 (1989).

[100] Gauthier, J. P., Hammouri, H., and Othman, S., "A Simple Observer for Nonlinear Systems Applications to Bioreactors," IEEE Transactions on Automatic Control, Vol 37, pp. 875-880 (1992).

[101] Soroush, M., "Nonlinear State-Observer Design with Application to Reactors," Chemical Engineering Science, Vol 52, pp. 387-404 (1997).

[102] Ciccarella, G., Dalla Mora, M., and Germani, A., "A Luenberger-Like Observer for Nonlinear Systems," International Journal of Control, Vol 57, pp. 537-556 (1993).

[103] Valluri, S., and Soroush, M., "Nonlinear State Estimation in the Presence of Multiple Steady States," Industrial and Engineering Chemistry Research, Vol 35, pp. 2645-2659 (1996).

[104] Baumann, W. T., and Rugh, W. J., "Feedback Control of Nonlinear Systems by Extended Linearization," IEEE Transactions on Automatic Control, Vol 31, pp. 40-46 (1986).

[105] Walcott, B. L., and Zak, S. H., "Observation of Dynamical Systems in the Presence of Bounded Nonlinearities/Uncertainties," Proceedings of the Twenty-fifth IEEE Conference on Decision and Control, Athens, Greece, pp. 961-966 (1986).

[106] Tsinias, J., "Further Results on the Observer Design Problem," Systems and Control Letters, Vol 14, pp. 411-418 (1990).

[107] Walcott, B. L., Corless, M. J., and Zak, S. H., "Comparative Study of Non-linear State-Observation Techniques," International Journal of Control, Vol 45, pp. 2109-2132 (1987).

[108] Hammett, K. D., Hall, C. D., and Ridgely, D. B., "Controllability Issues in Nonlinear State-Dependent Riccati Equation Control," Journal of Guidance, Control and Dynamics, Vol 21, pp. 767-773 (1998).

[109] Brogan, W. L., *Modern Control Theory*, 3rd ed., Englewood Cliffs, New Jersey: Prentice Hall, 1991.

[110] Banks, H. T., del Rosario, R. C. H., and Smith, R. C., "Reduced Order Model Feedback Control Design: Computational Studies for Thin Cylindrical Shells," CRSC Technical Report CRSC-TR98-25, N.C. State University (1998).

[111] Banks, H. T., del Rosario, R. C. H., and Smith, R. C., "Reduced Order Model Feedback Control Design: Numerical Implementation in a Thin Shell Model," CRSC Technical Report CRSC-TR98-27, N.C. State University (1998), and IEEE Transactions on Automatic Control, to appear.