Hessian-Based Model Reduction with Applications to Initial-Condition Inverse Problems

by

Omar Shahid Bashir

Submitted to the Department of Aeronautics and Astronautics
in partial fulfillment of the requirements for the degree of

Master of Science in Aerospace Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2007

© Massachusetts Institute of Technology 2007. All rights reserved.

Author . . . .
Department of Aeronautics and Astronautics
August 23, 2007

Certified by . . . .
Karen E. Willcox
Associate Professor of Aeronautics and Astronautics
Thesis Supervisor

Accepted by . . . .
David L. Darmofal
Associate Professor of Aeronautics and Astronautics
Chair, Committee on Graduate Students

Hessian-Based Model Reduction with Applications to Initial-Condition Inverse Problems

by

Omar Shahid Bashir

Submitted to the Department of Aeronautics and Astronautics
on August 23, 2007, in partial fulfillment of the
requirements for the degree of
Master of Science in Aerospace Engineering

Abstract

Reduced-order models that are able to approximate output quantities of interest of high-fidelity computational models over a wide range of input parameters play an important role in making tractable large-scale optimal design, optimal control, and inverse problem applications. We consider the problem of determining a reduced model of an initial value problem that spans all important initial conditions, and pose the task of determining appropriate training sets for reduced-basis construction as a sequence of optimization problems.

We show that, under certain assumptions, these optimization problems have an explicit solution in the form of an eigenvalue problem, yielding an efficient Hessian-based model reduction algorithm that scales well to systems with states of high dimension. Furthermore, tight upper bounds are given for the error in the outputs of the reduced models. The reduction methodology is demonstrated for several linear systems, including a large-scale contaminant transport problem.

Models constructed with the Hessian-based approach are used to solve an initial-condition inverse problem, and the resulting initial condition estimates compare favorably to those computed with high-fidelity models and low-rank approximations. Initial condition estimates are then formed with limited observational data to demonstrate that predictions of system state using reduced models are possible given relatively short measurement time windows. We show that reduced state can be used to approximate full state given an appropriate reduced basis, meaning that approximate forward simulations of large-scale systems can be computed in reduced space.

Thesis Supervisor: Karen E. Willcox
Title: Associate Professor of Aeronautics and Astronautics

Acknowledgements

My advisor, Prof. Karen Willcox, has provided so much support with her extensive technical knowledge and her words of encouragement. She has always been completely accessible even when we're on different continents. I especially want to thank her for helping me to take advantage of every opportunity, and for doing so with my best interests in mind. I am also thankful to Prof. Omar Ghattas for his advice and for allowing me to absorb some of his insight into inverse problems and related topics. I thank Bart van Bloemen Waanders and Judy Hill for answering all my questions and for making my stay at Sandia productive and enjoyable.

My friends in the ACDL were important during the past two years: Tan Bui brought me up to speed in many areas when I was just starting my research, and Garrett Barter was always willing to help as well. JM, Leia, Laslo, David, Josh, Alejandra, Theresa, and Tudor: sorry for bringing Settlers in and damaging your good work habits. It has been fun working with you guys and with all the other students in the lab. I'm also grateful to my roommate Adam Kumpf for asking hard questions about my research when I was stuck. Finally, I thank God for giving me parents who work hard to support me in all my academic pursuits.

This work was made possible by a National Science Foundation Graduate Research Fellowship. It was also supported by the NSF under DDDAS grant CNS-0540186 (program director Dr. Frederica Darema), the Air Force Office of Scientific Research (program manager Dr. Fariba Fahroo), and the Computer Science Research Institute at Sandia National Laboratories.

Contents

1 Introduction **15**

1.1 Motivation 15

1.2 Objectives 16

1.3 Previous Work 17

1.4 Overview 18

2 Model Reduction Framework **19**

2.1 Reduction via Projection 19

2.2 Proper Orthogonal Decomposition 23

2.3 Output-Weighted POD for Time-Dependent Problems 24

3 Hessian-Based Construction of Reduced-Order Models **27**

3.1 Theoretical Approach 27

3.2 Error Analysis 33

3.3 Large Scale Implementation 34

4 Application: Convection-Diffusion Transport Problem **37**

4.1 Two-dimensional model problem 37

    4.1.1 Problem description 37

    4.1.2 Reduced model performance 40

    4.1.3 Ten-sensor case 46

    4.1.4 Observations and Recommendations 48

    4.1.5 Output-Weighted POD Results 50

4.2 Contaminant Transport in a 3-D Urban Canyon 51

5 Application: Measurement, Inversion, and Prediction 57

5.1 Estimating the Initial Condition 58

5.1.1 Low-Rank Hessian Approximation 60

5.1.2 Reduced-Order Inverse Problem Solution 61

5.1.3 Inverse Problem Results 62

5.1.4 Implementation Cost Comparison 70

5.2 Using Reduced State to Approximate Full State 72

5.2.1 Comparison of Full State to its Approximation 72

5.2.2 State Approximation in the Inverse Problem Context 74

5.3 State Prediction from Limited Observational Data 76

5.3.1 Time-Limited Estimates of the Initial Condition 78

5.3.2 Time-Limited Inversion Results 79

6 Conclusion 83

6.1 Summary 83

6.2 Future Work 84

List of Figures

4-1 The computational domain and locations of sensor output nodes. Top: two-sensor case, bottom: ten-sensor case. 38

4-2 A comparison of the Hessian eigenvalue spectra of H for the two- and ten-output cases. Pe = 100. 40

4-3 Sample test initial conditions used to compare reduced model outputs to full-scale outputs. 42

4-4 Top: Maximum error, ε_{max}, for reduced models computed using Algorithms 1 and 2. Bottom: Error for test initial condition (b), ε. 44

4-5 A comparison of full ($N = 1860$) and reduced ($n = 196$) outputs for two-sensor case using test initial condition (b). Pe=1000, $\bar{\lambda} = 0.01$, $\bar{\mu} = 10^{-4}$, $\varepsilon = 0.0039$. 44

4-6 A comparison between full and reduced solutions at sensor location 1 for two different values of $\bar{\mu}$. Test initial condition (a) was used to generate the data. Pe=10, $\bar{\lambda} = 0.01$, two-sensor case. The output at the first sensor location is plotted here. 45

4-7 Lowering $\bar{\lambda}$ to increase p, the number of Hessian eigenvector initial conditions used in basis formation, leads to more accurate reduced-order output. Test initial condition (c) was used with two output sensors, Pe=100 and $\bar{\mu} = 10^{-4}$. The output at the second sensor location is plotted here. 46

4-8 A comparison of the full ($N = 1860$) and reduced ($n = 245$) outputs for all $Q = 10$ locations of interest. Test initial condition (c) was used to generate these data with Pe = 100, $\bar{\mu} = 10^{-4}$, $\bar{\lambda} = 0.1$. 47

4-9 A measure of the error in six different reduced models of the same system plotted versus their sizes n for the ten-sensor case. The three plots were generated with test initial conditions (a), (b), and (c), respectively. Pe=100, $Q = 10$ outputs. 49

4-10 Building geometry and locations of outputs for the 3-D urban canyon problem. 53

4-11 Transport of contaminant concentration through urban canyon at six different instants in time, beginning with the initial condition shown in upper left. 54

4-12 Full (65,600 states) and reduced (137 states) model contaminant concentration predictions at each of the six output nodes for the three-dimensional urban canyon example. 55

5-1 The actual initial condition used for the experiments; the goal of solving the inverse problem is to find this distribution from sensor measurements. Formed by superposing 10 Gaussian distributions with random centers and standard deviations. 63

5-2 The first 150 eigenvalues in the spectra of $H + \beta I$ and $H_r + \beta I$ with $\beta = 0$ and $\beta = 0.001$. The reduced model used to construct H_r is the baseline model with $\bar{\lambda} = 0.1$, $\bar{\mu} = 10^{-4}$, and $n = 245$. $Pe = 100$, $Q = 10$ outputs. 65

5-3 Error in low-rank and reduced-order initial condition versus actual initial condition x_0^a (top) and versus truth initial condition x_0^t (bottom). Data from reduced models formed with baseline values of $\bar{\lambda} = 0.1$ and $\bar{\mu} = 10^{-4}$ and with strict values of $\bar{\lambda} = 0.01$ and $\bar{\mu} = 10^{-6}$ are shown. Unless otherwise stated, the regularization constant for all trials is $\beta = 0.001$. $Pe = 100$, $Q = 10$ outputs. 67

5-4 The truth initial condition estimate x_0^t (top) and the reduced-order estimate x_0^{rom}. The reduced model used is the baseline model with $\bar{\lambda} = 0.1$, $\bar{\mu} = 10^{-4}$, and $n = 245$. $Pe = 100$, $Q = 10$ outputs. 68

5-5 The the actual sensor measurements (outputs) compared to $\mathbf{y}$ and $\mathbf{y}_r$. Here, $\mathbf{y}$ is generated by solving forward in time with the full-order equations starting from truth initial condition x_0^t; $\mathbf{y}_r$ is generated by solving forward in time with the reduced-order equations starting from reduced initial condition x_0^{rom}. Both initial condition estimates are shown in Figure 5-4. 69

5-6 Full-state approximation error with three different reduced-order models with identical initial conditions. $Pe = 100$, $Q = 10$ outputs. 73

5-7 State at $t = 0.4$ as calculated by the high-fidelity (top) and by the reduced-order systems of equations (middle), along with the error between the two snapshots (bottom). The reduced model ($n = 361$) with strict $\bar{\mu}$ was used for this comparison. $Pe = 100$, $Q = 10$ outputs. . . 75

5-8 At right, full-state approximation error with respect to high-fidelity forward solve starting from the truth initial condition x_0^t. The data are shown for three different reduced-order models with their respective initial conditions x_0^{rom}. On the left, the same data plotted against the actual state evolution in the domain, including the truth case for comparison. $Pe = 100$, $Q = 10$ outputs. 76

5-9 The actual state x^a in the domain at $t = 0.2$ (top); the state x^t as calculated by the high-fidelity system of equations beginning from the truth initial condition x_0^t (middle); and the state x^{rom} as calculated by the reduced-order system of equations beginning from the reduced-order initial condition x_0^{rom} (bottom). The reduced model ($n = 361$) with strict $\bar{\mu}$ was used for this comparison. Note the fine grid associated with the actual state. $Pe = 100$, $Q = 10$ outputs. 77

5-10 Effect of varying the length of observation time on the error between estimated initial condition and actual initial condition. The baseline and strict reduced-order models are described in Section 5.1.3. $Pe = 100$, $Q = 10$ outputs. 80

5-11 Full-order (truth) and reduced order estimates $\tilde{x}_0^t$ and $\tilde{x}_0^{rom}$ of the initial condition given a time window of length $t_f = 0.2$. Compare to Figure 5-4, which shows inversion results using a much longer time horizon of $t_f = 1.4$. The reduced-order model used is the baseline model with $\bar{\lambda} = 0.1$, $\bar{\mu} = 10^{-4}$, and $n = 245$. $Pe = 100$, $Q = 10$ outputs. 82

List of Tables

4.1 Properties of various reduced-order models of a full-scale system with Pe=100 and two output sensors. The errors ε and ε_{max} are defined in (3.20) and (3.21), respectively; ε is evaluated when each reduced system (of dimension n) is subjected to test initial condition (c). 41

4.2 Properties of various reduced-order models of a full-scale system with Pe=1000 and two output sensors. The errors ε and ε_{max} are defined in (3.20) and (3.21), respectively; ε is evaluated when each reduced system (of dimension n) is subjected to test initial condition (a). 43

5.1 Summary of notation used to represent initial condition quantities. . 58

Chapter 1

Introduction

1.1 Motivation

Reduced-order models that are able to approximate outputs of high-fidelity computational models over a wide range of input parameters have an important role to play in making tractable large-scale optimal design, optimal control, and inverse problem applications. For example, consider a time-dependent inverse problem in which the goal is to estimate the initial condition. The constraints are the state equations describing the dynamics of the system, and the objective is the difference between the measured state observations and observations predicted by the state equations starting from the unknown initial condition.

When the physical system being simulated is governed by partial differential equations in three spatial dimensions and time, the forward problem alone (i.e. solution of the PDEs for a given initial condition) may require many hours of supercomputer time. The inverse problem, which requires repeated solution of the forward problem, may then be out of reach in situations where rapid assimilation of the data is required. In particular, when the simulation is used as a basis for forecasting or decision-making, a reduced model that can execute much more rapidly than the high-fidelity PDE simulation is needed. A crucial requirement for the reduced model is that it be able to replicate the output quantities of interest (i.e. the observables) of the PDE simulation over a wide range of initial conditions, so that it may serve as a surrogate of the high-fidelity PDE simulation during inversion.

One popular method for generating a reduced model is through a projection basis; for example, by proper orthogonal decomposition (POD) in conjunction with the method of snapshots. To build such a reduced order model, one typically constructs a training set by sampling the space of (discretized) initial conditions. When this space is high-dimensional, the problem of adequately sampling it quickly becomes intractable. Fortunately, for many ill-posed inverse problems, many components of the initial condition space have minimal or no effect on the output observables. This is particularly true when the observations are sparse. In this case, it is likely that an effective reduced model can be generated with few sample points. If appropriate sample points can be located, it is possible to form a reduced model which can accept a wide range of initial conditions while providing accurate outputs. The initial-condition sampling issue is therefore the central theme of this thesis.

1.2 Objectives

The primary objectives of this thesis are:

- to present a model reduction approach that is capable of generating reduced models that provide accurate output replication for a wide range of possible initial conditions;
- to explain the sources of computational cost associated with the new approach;
- to evaluate the approach by forming reduced models of high-fidelity linear systems and subjecting both full and reduced models to a variety of test initial conditons;
- to demonstrate that the resulting reduced models are well-suited for use in the efficient solution of initial-condition inverse problems and in the state estimation which often follows in practice.

1.3 Previous Work

Reduction techniques for large-scale systems have generally focused on a projection framework that utilizes a reduced-space basis. Methods to compute the basis in the large-scale setting include Krylov-subspace methods [18, 19, 24], approximate balanced truncation [25, 32, 33, 36], and proper orthogonal decomposition [16, 29, 35]. Progress has been made in development and application of these methods to optimization applications with a small number of input parameters, for example optimal control [2, 5, 28, 30] and parametrized design of interconnect circuits [15]. In the case of a high-dimensional input parameter space, the computational cost of determining the reduced basis by these techniques becomes prohibitive unless some sparse sampling strategy is employed.

For initial-condition problems of moderate dimension, a reduction method has been proposed that truncates a balanced representation of the finite-dimensional Hankel operator [17]. In [14], POD was used in a large-scale inverse problem setting to define a reduced space for the initial condition in which to solve the data assimilation problem. In that work, only a single initial condition was used to generate the state solutions necessary to form the reduced basis: either the true initial condition, which does contain the necessary information but would be unavailable in practice, or the background estimate of the initial state, which defines a forecast trajectory that may not be sufficiently rich in terms of state information.

For model reduction of linear time-invariant systems using multipoint rational Krylov approximations, two methods have recently been proposed to choose sample locations: an iterative method to choose an optimal set of interpolation points [26], and a heuristic statistically-based resampling scheme to select sample points [34]. To address the more general challenge of sampling a high-dimensional parameter space to build a reduced basis, the greedy algorithm was introduced in [38]. The key premise of the greedy algorithm is to adaptively choose samples by finding the location in parameter space where the error in the reduced model is maximal. In [37], the greedy algorithm was applied to find reduced models for the parametrized steady incompressible Navier-Stokes equations. In [22, 23], the greedy algorithm was combined with *a posteriori* error estimators for parametrized parabolic partial differential equations, and applied to several optimal control and inverse problems.

1.4 Overview

In this thesis, we present a new methodology that employs an efficient sampling strategy to make tractable the task of determining reduced-order models for large-scale linear initial value problems. Reduced models formed using the Hessian-based methodology provide outputs which are similar to those computed with high-fidelity models. This accurate output replication over a wide range of initial conditions allows the reduced models to be used as surrogates for full-order models in initial-condition inverse problems.

Chapter 2 describes the projection framework used to derive the reduced-order dynamical system. It also provides details of the POD and proposes a new POD variant. We present in Chapter 3 the theoretical approach, leading to the Hessian-based reduction methodology. In Chapter 4, we first demonstrate the efficacy of the approach via numerical experiments on a problem of 2-D convective-diffusive transport. We also present an application to model reduction for 3-D contaminant transport in an urban canyon. Chapter 5 contains a demonstration of reduced-order inverse problem solution. It then illustrates the utility of reduced models for state estimation and prediction. We conclude the thesis with Chapter 6.

Chapter 2

Model Reduction Framework

This chapter first describes the reduced-order models intended for use in place of high-fidelity or full-order systems. It then summarizes in Section 2.2 the details of the proper orthogonal decomposition (POD) used throughout this work to generate reduced basis vectors. Finally, a new, output-dependent variant of POD is presented in Section 2.3.

2.1 Reduction via Projection

Consider the general linear discrete-time system

$$Ex(k+1) = Ax(k) + Bu(k), \qquad k = 0, 1, \ldots, T-1, \tag{2.1}$$

$$y(k) = Cx(k), \qquad k = 0, 1, \ldots, T, \tag{2.2}$$

with initial condition

$$x(0) = x_0, \tag{2.3}$$

where $x(k) \in \mathbb{R}^N$ is the system state at time t_k, the vector x_0 contains the specified initial state, and we consider a time horizon from $t = 0$ to $t = t_f$. The vectors $u(k) \in \mathbb{R}^P$ and $y(k) \in \mathbb{R}^Q$ contain, respectively, the P system inputs and Q system outputs at time t_k. In general, we are interested in systems of the form (2.1)–(2.3) that result from spatial and temporal discretization of PDEs. In this case, the dimension of the system, N, is very large and the matrices $E \in \mathbb{R}^{N\times N}$, $A \in \mathbb{R}^{N\times N}$, $B \in \mathbb{R}^{N\times P}$, and $C \in \mathbb{R}^{Q\times N}$ result from the chosen spatial and temporal discretization methods.

A reduced-order model of (2.1)–(2.3) can be derived by assuming that the state $x(k)$ is represented as a linear combination of n basis vectors,

$$\hat{x}(k) = V x_r(k), \tag{2.4}$$

where $\hat{x}(k) \in \mathbb{R}^N$ is the reduced model approximation of the state $x(k)$ and $n \ll N$. The projection matrix $V \in \mathbb{R}^{N\times n}$ contains as columns the orthonormal basis vectors v_i, i.e., $V = [v_1 \ v_2 \ \cdots \ v_n]$, and the reduced-order state $x_r(k) \in \mathbb{R}^n$ contains the corresponding modal amplitudes for time t_k. Using the representation (2.4) together with a Galerkin projection of the discrete-time system (2.1)–(2.3) onto the space spanned by the basis V yields the reduced-order model with state x_r and output y_r,

$$\begin{aligned} E_r x_r(k+1) &= A_r x_r(k) + B_r u(k), \qquad k = 0, 1, \ldots, T-1, && (2.5)\\ y_r(k) &= C_r x_r(k), \qquad k = 0, 1, \ldots, T, && (2.6)\\ x_r(0) &= V^T x_0, && (2.7) \end{aligned}$$

where $E_r = V^T E V$, $A_r = V^T A V$, $B_r = V^T B$, and $C_r = CV$.

Since the system (2.1)–(2.3) is linear, the effects of inputs u and initial conditions x_0 can be considered separately. In this thesis, we focus on the initial-condition problem and, without loss of generality, assume that $u(k) = 0, \ k = 0, 1, \ldots, T-1$. For convenience of notation, we write the discrete-time system (2.1)–(2.3) in matrix form as

$$\begin{aligned} \mathbf{A}\mathbf{x} &= \mathbf{F} x_0, && (2.8)\\ \mathbf{y} &= \mathbf{C}\mathbf{x}, && (2.9) \end{aligned}$$

where

$$\mathbf{x} = \begin{bmatrix} x(0) \\ x(1) \\ \vdots \\ x(T) \end{bmatrix}, \qquad \mathbf{y} = \begin{bmatrix} y(0) \\ y(1) \\ \vdots \\ y(T) \end{bmatrix}. \tag{2.10}$$

The matrices $\mathbf{A} \in \mathbb{R}^{N(T+1)\times N(T+1)}$, $\mathbf{F} \in \mathbb{R}^{N(T+1)\times N}$, and $\mathbf{C} \in \mathbb{R}^{Q(T+1)\times N(T+1)}$ in (2.8) and (2.9) are given by

$$\mathbf{A} = \begin{bmatrix} E & 0 & \cdots & \cdots & 0 \\ -A & E & 0 & & \\ 0 & -A & E & \ddots & \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & & 0 & -A & E \end{bmatrix}, \mathbf{F} = \begin{bmatrix} E \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \mathbf{C} = \begin{bmatrix} C & 0 & \cdots & \cdots & 0 \\ 0 & C & 0 & & \\ \vdots & 0 & C & \ddots & \\ \vdots & & \ddots & \ddots & 0 \\ 0 & & & 0 & C \end{bmatrix}. \tag{2.11}$$

Similarly, the reduced-order model (2.5)–(2.7) can be written in matrix form as

$$\mathbf{A}_r \mathbf{x}_r = \mathbf{F}_r x_0, \tag{2.12}$$

$$\mathbf{y}_r = \mathbf{C}_r \mathbf{x}_r, \tag{2.13}$$

where $\mathbf{x}_r$ and $\mathbf{y}_r$ are defined analogously to $\mathbf{x}$ and $\mathbf{y}$ as

$$\mathbf{x}_r = \begin{bmatrix} x_r(0) \\ x_r(1) \\ \vdots \\ x_r(T) \end{bmatrix}, \qquad \mathbf{y}_r = \begin{bmatrix} y_r(0) \\ y_r(1) \\ \vdots \\ y_r(T) \end{bmatrix}. \tag{2.14}$$

The matrices $\mathbf{A}_r \in \mathbb{R}^{n(T+1)\times n(T+1)}$, $\mathbf{F}_r \in \mathbb{R}^{n(T+1)\times N}$, and $\mathbf{C}_r \in \mathbb{R}^{Q(T+1)\times n(T+1)}$ are given by

$$\mathbf{A}_r = \begin{bmatrix} E_r & 0 & \cdots & \cdots & 0 \\ -A_r & E_r & 0 & & \\ 0 & -A_r & E_r & \ddots & \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & & 0 & -A_r & E_r \end{bmatrix}, \mathbf{F}_r = \begin{bmatrix} E_r V^T \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \tag{2.15}$$

$$\mathbf{C}_r = \begin{bmatrix} C_r & 0 & \cdots & \cdots & 0 \\ 0 & C_r & 0 & & \\ \vdots & 0 & C_r & \ddots & \\ \vdots & & \ddots & \ddots & 0 \\ 0 & & & 0 & C_r \end{bmatrix}.$$

As an alternative to the discrete-time representation of a system (2.1)–(2.2), a continuous representation of the form

$$\mathcal{M}\dot{x}(k) = \mathcal{A}x(k) + \mathcal{B}u(k), \tag{2.16}$$

$$y(k) = Cx(k), \tag{2.17}$$

where $\dot{x}$ is the vector of state derivatives with respect to time, may be available. Here, $\mathcal{M} \in \mathbb{R}^{N\times N}$, $\mathcal{A} \in \mathbb{R}^{N\times N}$, and $\mathcal{B} \in \mathbb{R}^{N\times P}$ might come from a finite-element discretization with mass matrix $\mathcal{M}$ and stiffness matrix $\mathcal{A}$. We can arrive at the matrix representation (2.8)–(2.9) in this case by choosing a temporal discretization method and timestep Δt, again assuming that $u(k) = 0, k = 0, 1, \ldots, T-1$ without loss of generality. For example, if the Crank-Nicolson method is chosen, the matrices in (2.11) with

$$E = \mathcal{M} - \frac{1}{2}\Delta t\mathcal{A}, \tag{2.18}$$

$$A = \mathcal{M} + \frac{1}{2}\Delta t\mathcal{A}, \tag{2.19}$$

can be used in conjunction with (2.8)–(2.9) to solve for system state over time. Note that $\mathbf{C}$ remains as given in (2.11) since the input-to-output mapping is unchanged.

To form the reduced-order system of equations in this continuous case, the Galerkin projection is applied to $\mathcal{M}$ and $\mathcal{A}$ or, equivalently, to E and A:

$$\begin{aligned} E_r &= V^T E V, && (2.20)\\ A_r &= V^T A V. && (2.21) \end{aligned}$$

These matrices form the block entries in (2.15); the reduced state can be obtained by solving (2.12).

In many cases, we are interested in rapid identification of initial conditions from sparse measurements of the states over a time horizon; we thus require a reduced-order model that will provide accurate outputs for any initial condition contained in some set $\mathcal{X}_0$. Using the projection framework described above, the task therefore becomes one of choosing an appropriate basis V so that the error between full-order output $\mathbf{y}$ and the reduced-order output $\mathbf{y}_r$ is small for all initial conditions of interest.

2.2 Proper Orthogonal Decomposition

To compute the basis V via POD, a sample set of initial conditions must first be chosen. At each selected initial condition, a forward simulation is performed to generate a set of states, commonly referred to as snapshots. The POD is then applied to the snapshot set in order to form reduced basis vectors. Although the choice of sample initial conditions is the focus of Chapter 3, we describe the process of forming basis vectors using the method of snapshots [35] in this section.

Assume that we wish to construct a reduced basis with data from p forward solves of the high-fidelity system with different initial conditions. The instantaneous state solutions $x_i^j \in \mathbb{R}^{N\times 1}$, $i = 0, 1, \ldots, T$, $j = 1, 2, \ldots, p$ are collected in the snapshot matrix $X \in \mathbb{R}^{N\times p(T+1)}$.

The POD basis vectors $v_1, v_2, \ldots, v_n$ are chosen to be the orthonormal set that best represent the given snapshots in X [9]. This occurs when the n basis vectors are taken to be the n most dominant eigenvectors of XX^T or, equivalently, the first n left singular vectors of X.

The POD minimizes the quantity

$$\hat{E} = \sum_{i=0}^{T+1} \sum_{j=1}^{p} \left[x_i^j - VV^T x_i^j\right]^T \left[x_i^j - VV^T x_i^j\right], \tag{2.22}$$

for a fixed number of basis vectors. $\hat{E}$ represents the error between the original snapshots and their representations in the reduced space. This error is also equal to the sum of the squares of the singular values corresponding to the singular vectors not included in the basis,

$$\hat{E} = \sum_{k=n+1}^{p(T+1)} \sigma_k^2, \tag{2.23}$$

where σ_k is the k^{th} singular value of X.

2.3 Output-Weighted POD for Time-Dependent Problems

In the classical POD described in Section 2.2, the basis vectors are chosen such that they minimize the *state* error in a least-squares sense. However, the outputs do not influence the basis construction as they do in the weighted POD method we describe in this section.

The idea of weighting snapshots to improve basis quality has been explored before [13]. It has been emphasized that weighting the snapshots can have a significant impact on the selection of dominant modes [21]. One such implementation involves an integer weighting scale such that multiple copies of strongly-weighted snapshots are included in the data set before POD is applied [11].

Regardless of how the weights are calculated, we first summarize the approach used to form POD basis vectors using these weights [13]. First, an ensemble average of the data $\bar{x} \in \mathbb{R}^N$ is computed using a linear combination of the snapshots $x(i)$. If, for simplicity, we are only interested in the $T+1$ snapshots from a single forward solve, then

$$\bar{x} = \sum_{i=0}^{T} \omega_i x(i), \tag{2.24}$$

where the snapshot weights ω_i are chosen such that $0 < \omega_i < 1, \sum_{i=0}^{T} \omega_i = 1$. A modified snapshot matrix $\check{X}$ can then be computed using the ensemble average:

$$\check{X} = \left[x(0) - \bar{x}, x(1) - \bar{x}, \ldots, x(T) - \bar{x} \right]. \tag{2.25}$$

If W is a defined as a diagonal matrix of weights ω_0 to ω_T, then the weighted POD basis vectors are the eigenvectors $\check{v}_i$ in the eigenvalue problem

$$\check{X} W \check{X}^T \check{v}_i = \check{\sigma}_i \check{v}_i, \qquad i = 1, 2, \ldots, N, \tag{2.26}$$

where $\check{\sigma}_i$ are the eigenvalues of $\check{X} W \check{X}^T$.

We propose a new choice of weights which takes into account the current and future *output* response associated with each snapshot. The motivation behind this choice is the assumption that the least valuable snapshots for reduced basis formation are those which, when used as initial conditions, cause little or no output response. Those which cause a relatively large output over time are considered the most important snapshots. This distinction allows us to form a quantitative weighting method. Specifically, the non-normalized weights $\hat{\omega}_i$ are given by

$$\hat{\omega}_i = \int_{t_i}^{t_f} ||y(t)||_2^2 \, dt, \qquad i = 0, 1, \ldots, T, \tag{2.27}$$

where t_i is the time corresponding to snapshot i, and t_f is the final time at which snapshots are collected. The output vector $y(t) \in \mathbb{R}^Q$ can be calculated via the state-to-output mapping matrix C. To normalize the weights,

$$\omega_i = \frac{\hat{\omega}_i}{\hat{\omega}_{max}}, \tag{2.28}$$

where $\hat{\omega}_{max}$ is the maximum non-normalized weight in the snapshot set. Since the algorithms presented in Chapter 3 require forward solves for many different initial conditions, the snapshot sets for each of these initial conditions are kept separate so that weights are calculated independently within each set; however, normalizing every snapshot by the largest $\hat{\omega}_{max}$ across all snapshot sets may also be a valid approach. The former method is used in this work.

Since all snapshots x are already collected before the calculation of weights, the only additional costs involved in finding each weight are the single matrix-vector multiplication $y = Cx$ and a numerical integration to approximate the continuous-time integral in (2.27). The intended benefit of the output-weighted POD is the construction of bases which are similar in size to those created with classical POD but which provide a higher degree of output accuracy when both are subjected to test initial conditions. Alternatively, the method can be viewed as an attempt to reduce the number of basis vectors needed to achieve a certain level of accuracy. The performance of the method is discussed in Section 4.1.5.

Chapter 3

Hessian-Based Construction of Reduced-Order Models

In this chapter, a methodology to determine a basis that spans the space of important initial conditions is presented. It has been shown that in the case of systems that are linear in the state, POD is equivalent to balanced truncation if the snapshots are computed for all possible initial conditions [31]. Since sampling all possible initial conditions is not feasible for large-scale problems, we propose an adaptive approach to identify important initial conditions that should be sampled. The approach is motivated by the greedy algorithm of [38], which proposed an adaptive approach to determine the parameter locations at which samples are drawn to form a reduced basis. For the linear finite-time-horizon problem considered here, we show that the greedy algorithm can be formulated as an optimization problem that has an explicit solution in the form of an eigenvalue problem.

3.1 Theoretical Approach

Our task is to find an appropriate reduced basis and associated reduced model: one that provides accurate outputs for all initial conditions of interest. We define an optimal basis, V^*, to be one that minimizes the maximal L_2 error between the full-order and reduced-order outputs of the fully discrete system over all admissible initial conditions,

$$V^* = \arg\min_{V} \max_{x_0 \in \mathcal{X}_0} \ (\mathbf{y} - \mathbf{y}_r)^T (\mathbf{y} - \mathbf{y}_r) \tag{3.1}$$

$$\begin{aligned} \text{where} \qquad \mathbf{A}\mathbf{x} &= \mathbf{F}x_0, && (3.2)\\ \mathbf{y} &= \mathbf{C}\mathbf{x}, && (3.3)\\ \mathbf{A}_r\mathbf{x}_r &= \mathbf{F}_r x_0, && (3.4)\\ \mathbf{y}_r &= \mathbf{C}_r\mathbf{x}_r. && (3.5) \end{aligned}$$

For this formulation, the only restriction that we place on the set $\mathcal{X}_0$ is that it contain vectors of unit length. This prevents unboundedness in the optimization problem, since otherwise the error in the reduced system could be made arbitrarily large. Naturally, because the system is linear, the basis V^* will still be valid for initial conditions of any finite norm.

A suboptimal but computationally efficient approach to solving the optimization problem (3.1)–(3.5) is inspired by the greedy algorithm of [38]. Construction of a reduced basis for a steady or unsteady problem with parameter dependence, as considered in [37, 22], requires a set of snapshots, or state solutions, over the parameter–time space. The greedy algorithm adaptively selects these snapshots by finding the location in parameter–time space where the error between the full-order and reduced-order models is maximal, updating the basis with information gathered from this sample location, forming a new reduced model, and repeating the process. In the case of the initial-condition problem (3.1)–(3.5), the greedy approach amounts to sampling at the initial condition $x_0^* \in \mathcal{X}_0$ that *maximizes* the error in (3.1).

The key step in the greedy algorithm is finding the worst-case initial condition x_0^*, which we achieve by solving the modified optimization problem,

$$x_0^* = \arg\max_{x_0 \in \mathcal{X}_0} \ (\mathbf{y} - \mathbf{y}_r)^T (\mathbf{y} - \mathbf{y}_r) \tag{3.6}$$

$$\begin{aligned} \text{where} \qquad \mathbf{A}\mathbf{x} &= \mathbf{F}x_0, && (3.7)\\ \mathbf{y} &= \mathbf{C}\mathbf{x}, && (3.8)\\ \mathbf{A}_r\mathbf{x}_r &= \mathbf{F}_r x_0, && (3.9)\\ \mathbf{y}_r &= \mathbf{C}_r\mathbf{x}_r. && (3.10) \end{aligned}$$

Equations (3.6)–(3.10) define a large-scale optimization problem, which includes the full-scale dynamics (3.7), (3.8) as constraints. The approach taken in [37, 22] is to replace these constraints with error estimators, so that the full-scale model does not need to be invoked during solution of the optimization problem. Further, in [37, 22], the optimization problem (20)-(24) is solved by a grid-search technique that addresses problems associated with non-convexity and non-availability of derivatives.

In this chapter, we exploit the linearity of the state equations to eliminate the full-order and reduced-order states and yield an equivalent unconstrained optimization problem. Eliminating the constraints (3.7)–(3.10) by solving for the full and reduced states yields

$$x_0^* = \arg \max_{x_0 \in \mathcal{X}_0} \quad x_0^T H^e x_0, \tag{3.11}$$

where

$$H^e = \left(\mathbf{C}\mathbf{A}^{-1}\mathbf{F} - \mathbf{C}_r\mathbf{A}_r^{-1}\mathbf{F}_r\right)^T \left(\mathbf{C}\mathbf{A}^{-1}\mathbf{F} - \mathbf{C}_r\mathbf{A}_r^{-1}\mathbf{F}_r\right). \tag{3.12}$$

It can be seen that (3.11) is a quadratic unconstrained optimization problem with Hessian matrix $H^e \in \mathbb{R}^{N \times N}$. From (3.12), it can be seen that H^e is a symmetric positive semi-definite matrix that does not depend upon the state or initial condition. The eigenvalues of H^e are therefore non-negative. Since we are considering initial conditions of unit norm, the solution x_0^* maximizes the Rayleigh quotient; therefore, the solution of (3.11) is given by the eigenvector z_1^e corresponding to the largest eigenvalue λ_1^e of H^e:

$$H^e z_1^e = \lambda_1^e z_1^e. \tag{3.13}$$

The eigenvector z_1^e is the initial condition for which the error in reduced model output prediction is largest.

These ideas motivate the following basis-construction algorithm for the initial condition problem.

Algorithm 1 *Greedy Reduced Basis Construction*

Initialize with $V = 0$*, so that the initial reduced-order model is zero.*

1. *For the error Hessian matrix,* H^e *as defined in* (3.12)*, find the eigenvector* z_1^e *with largest eigenvalue* λ_1^e*.*
2. *Set* $x_0 = z_1^e$ *and compute the corresponding solution* $\mathbf{x}$ *using* (2.8)*.*
3. *Update the basis* V *by adding the new information from the snapshots* $x(k)$*,* $k = 0, 1, \dots, T$*.*
4. *Update the reduced model using the new basis and return to Step 1.*

In Step 3 of Algorithm 1, the basis could be computed from the snapshots, using, for example, the POD. A rigorous termination criterion for the algorithm is available in the form of an error bound, which will be discussed below. It should be noted that, while the specific form of Algorithm 1 applies only in the linear case, the greedy sampling concept is applicable to nonlinear problems. In the general nonlinear case, one would solve an optimization problem similar in form to (3.6)–(3.10), but with the appropriate nonlinear governing equations appearing as constraints. In this case, the explicit eigenvalue solution to the optimization problem would not hold; instead, one would use a method that is appropriate for large-scale simulation-constrained optimization (see [3]) to solve the resulting optimization problem.

Under certain assumptions, the form of H^e in (3.11) can be simplified, leading to an algorithm that avoids construction of the reduced model at every greedy iteration. We proceed by decomposing a general initial condition vector as

$$x_0 = x_0^V + x_0^\perp, \tag{3.14}$$

where x_0^V is the component of x_0 in the subspace spanned by the current basis V, and $x_0^\perp$ is the component of x_0 in the orthogonal complement of that subspace. Substituting (3.14) into the objective function (3.11), we recognize that $\mathbf{F}_r x_0^\perp = 0$, using the form of $\mathbf{F}_r$ given by (2.15) and that, by definition, $V^T x_0^\perp = 0$. The unconstrained optimization problem (3.11) can therefore be written as

$$\begin{aligned} x_0^* = \arg\max_{x_0 \in \mathcal{X}_0} \; & \left(\mathbf{CA}^{-1}\mathbf{F}x_0^V + \mathbf{CA}^{-1}\mathbf{F}x_0^\perp - \mathbf{C}_r\mathbf{A}_r^{-1}\mathbf{F}_r x_0^V\right)^T \\ & \left(\mathbf{CA}^{-1}\mathbf{F}x_0^V + \mathbf{CA}^{-1}\mathbf{F}x_0^\perp - \mathbf{C}_r\mathbf{A}_r^{-1}\mathbf{F}_r x_0^V\right). \end{aligned} \tag{3.15}$$

The expression (3.15) can be approximated by assuming that

$$\mathbf{CA}^{-1}\mathbf{F}x_0^V = \mathbf{C}_r\mathbf{A}_r^{-1}\mathbf{F}_r x_0^V, \tag{3.16}$$

which means that for initial conditions x_0^V in the space spanned by the basis, we assume that the reduced output exactly matches the full output, i.e. $\mathbf{y} = \mathbf{y}_r$. An approach to satisfying this condition will be described shortly. Using the approximation (3.16), we can rewrite (3.11) as

$$x_0^* = \arg\max_{x_0^\perp \in \mathcal{X}_0} \left(x_0^\perp\right)^T H x_0^\perp, \tag{3.17}$$

where

$$H = \left(\mathbf{CA}^{-1}\mathbf{F}\right)^T \left(\mathbf{CA}^{-1}\mathbf{F}\right). \tag{3.18}$$

$H \in \mathbb{R}^{N\times N}$ is now the Hessian matrix of the full-scale system, and does not depend on the reduced-order model. As before, H is a symmetric, positive semi-definite matrix that does not depend upon the state or initial condition.

If we choose to initialize the greedy algorithm with an empty basis, $V = 0$, then the maximizer of (3.17) on the first greedy iteration is given by the eigenvector of H corresponding to the largest eigenvalue. We denote this initial condition by z_1 and note that z_1 satisfies

$$Hz_1 = \lambda_1 z_1, \tag{3.19}$$

where λ_1 is the largest eigenvalue of H. We then set $V = z_1$. Under the assumption that (3.16) holds, on the second greedy iteration we would therefore seek the initial condition that maximizes (3.17). Clearly, this initial condition, which should be orthogonal to z_1, is given by z_2, the eigenvector of H corresponding to the second largest eigenvalue.

Returning to assumption (3.16), this condition can be satisfied if we include in the basis not just the sequence of optimal initial conditions $x_0^* = \{z_1, z_2, \ldots\}$, but rather the span of *all* snapshots (i.e. instantaneous state solutions contained in $\mathbf{x}$) obtained by solving (2.8) for each of the seed initial conditions $z_1, z_2, \ldots$. The approximation (3.16) will then be accurate, provided the final time t_f is chosen so that the output $y(k)$ is small for $k > T$. If the output is not small for $k > T$, then a snapshot collected at some time $t_{\bar{k}}$, where $\bar{k} < T$ but $\bar{k}$ is large, will be added to the basis; however, if that state were then used as an initial condition in the resulting reduced-order model, the resulting solution $\mathbf{y}_r$ would not necessarily be an accurate representation of $\mathbf{y}$. This is because the basis would not contain information about system state evolution after time $t_{T-\bar{k}}$. In that case, (3.16) would not hold. Further, by including both the initial conditions, z_i, and the corresponding snapshots, $\mathbf{x}$, in the basis, the sequence of eigenvectors z_i will no longer satisfy the necessary orthogonality conditions; that is, the second eigenvector z_2 may no longer be orthogonal to the space spanned by the basis comprising z_1 and its corresponding state solutions. This is because setting $x_0 = z_1$ and computing $\mathbf{x}$ will likely lead to some states that have components in the direction of z_2. We would therefore expect this simplification to be more accurate for the first few eigenvectors, and become less accurate as the number of seed initial conditions is increased.

These simplifications lead us to an alternate "one-shot" basis-construction algorithm for the initial condition problem. This algorithm does not solve the optimization problems (3.1)–(3.5) or (3.6)–(3.10) exactly, but provides a good approximate solution to the problem (3.6)–(3.10) under the conditions discussed above. We use the dominant eigenvectors of the Hessian matrix H to identify the initial-condition vectors that have the most significant contributions to the outputs of interest. These vectors are in turn used to initialize the full-scale discrete-time system to generate a set of state snapshots that are used to form the reduced basis.

Algorithm 2 *One-Shot Hessian-Based Reduced Basis Construction*

1. *For the full-order Hessian matrix, H as defined in* (3.18)*, find the p eigenvectors $z_1, z_2, \ldots, z_p$ with largest eigenvalues $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_p \geq \lambda_{p+1} \geq \ldots \geq \lambda_N \geq 0$.*
2. *For $i = 1, \ldots, p$, set $x_0 = z_i$ and compute the corresponding solution $\mathbf{x}^i$ using* (2.8)*.*
3. *Form the reduced basis as the span of the snapshots $x^i(k)$, $i = 1, 2, \ldots, p$, $k = 0, 1, \ldots, T$.*

Steps 2 and 3 in Algorithm 2 allow us to (approximately) satisfy the assumption (3.16) by including not just the initial conditions $z_1, z_2, \ldots, z_p$ in the basis but also the span of all snapshots generated from those initial conditions. The basis could be computed from the snapshots, using, for example, the POD.

3.2 Error Analysis

A direct measure of the quality of the reduced-order model is available using the analysis framework described above. We define the error, ε, due to a particular initial condition x_0 as

$$\varepsilon = ||\mathbf{y} - \mathbf{y}_r||_2 = \left|\left|\left(\mathbf{C}\mathbf{A}^{-1}\mathbf{F} - \mathbf{C}_r\mathbf{A}_r^{-1}\mathbf{F}_r\right) x_0\right|\right|_2 . \tag{3.20}$$

For a given reduced model, the dominant eigenvector of H^e provides the worst-case initial condition. Therefore, the value of the maximal error ε_{max} (for an initial condition of unit norm) is given by

$$\varepsilon_{max} = \sqrt{\lambda_1^e}, \tag{3.21}$$

where λ_1^e is the largest eigenvalue of the error Hessian H^e, defined by (3.12). The value ε_{max} provides both a measure on the quality of the reduced model and a quantitative termination criterion for the basis-construction algorithm.

In Algorithm 1, ε_{max} is readily available, and thus can be used to determine how many cycles of the algorithm to perform, i.e. the algorithm would be terminated when the worst-case error is sufficiently small. In Algorithm 2, it is computationally more efficient to select p, the number of seed initial conditions, based on the decay rate of the full Hessian eigenvalues $\lambda_1, \lambda_2, \ldots$ and to compute all the necessary eigenvectors $z_1, z_2, \ldots, z_p$ at once. Once the reduced model has been created using Algorithm 2, the error Hessian H^e can be formed and the error criterion (3.21) checked to determine if further sampling is required. While Algorithm 1 is expected to reduce the worst-case error more quickly, the one-shot Algorithm 2 is attractive since it depends only on the large-scale system properties and thus does not require us to build the reduced model on each cycle.

We also note that the eigenvectors of $H = (\mathbf{CA}^{-1}\mathbf{F})^T (\mathbf{CA}^{-1}\mathbf{F})$ are equivalent to the (right) singular vectors of $\mathbf{CA}^{-1}\mathbf{F}$. Since the latter quantity serves as an input-output mapping, use of its singular vectors for basis formation is intuitively attractive. It is also interesting to note that the Hessian H may be thought of as a finite-time observability Gramian [4].

3.3 Large Scale Implementation

We first discuss the implementation of Algorithm 2 in the large-scale setting, and then remark on the differences for Algorithm 1.

Algorithm 2 is a one-shot approach in which all of the eigenpairs can be computed from the single Hessian matrix H in (3.18). This matrix can be formed explicitly by first forming $\mathbf{A}^{-1}\mathbf{F}$, which requires N "forward solves" (i.e. solutions of forward-in-time dynamical systems with $\mathbf{A}$ as coefficient matrix), where N is the number of initial condition parameters; or else by first forming $\mathbf{A}^{-T}\mathbf{C}^T$, which requires Q "adjoint" solves (i.e. solutions of backward-in-time dynamical systems with $\mathbf{A}^T$ as coefficient matrix), where Q is the number of outputs. For large-scale problems with high-dimensional initial condition and output vectors, explicit formation and storage of H is thus intractable. (A similar argument can be made for the intractability of computing the singular value decomposition of $\mathbf{CA}^{-1}\mathbf{F}$.) Even if H could be formed and stored, computing its dominant spectrum would be prohibitive, since it is a dense matrix of order $N \times N$.

Instead, we use a matrix-free iterative method such as Lanczos to solve for the dominant eigenpairs of H. Such methods require at each iteration a matrix–vector product of the form Hw_k for some w_k, which is formed by successive multiplication of vectors with the component matrices that make up the Hessian in (3.18). At each iteration, this amounts to one forward and one adjoint solve involving the system $\mathbf{A}$. When the eigenvalues are well-separated, convergence to the largest eigenvalues of H is rapid. Moreover, when the spectrum decays rapidly, only a handful of eigenvectors are required by Algorithm 2. Many problems have Hessian matrices that are of low rank and spectra that decay rapidly, stemming from the limited number of initial conditions that have a significant effect on outputs of interest. For such problems the number of Lanczos iterations required to extract the dominant part of the spectrum is often independent of the problem size N.

Under this assumption, we can estimate the cost of Step 1 of Algorithm 2 (which dominates the cost) in the case when the dynamical system (2.8)–(2.9) stems from a discretized parabolic PDE. The cost of each implicit time step of a forward or adjoint solve is usually linear or weakly superlinear in problem size, using modern multilevel preconditioned linear solvers. Therefore for T time steps, overall work for a forward or adjoint solve scales as $TN^{1+\alpha}$, with α usually very small. For a 3-D spatial problem, a number of time steps on the order of the diameter of the grid, and an optimal preconditioner, this gives $\mathcal{O}(N^{4/3})$ complexity per forward solve, and hence per Lanczos iteration. Assuming the number of Lanczos iterations necessary to extract the dominant part of the spectrum is independent of the grid size, the overall complexity remains $\mathcal{O}(N^{4/3})$. (Compare this with straightforward formation of the Hessian and computation of the eigenvalues with the QR algorithm, which requires

$\mathcal{O}(N^3)$ work.)

Algorithm 1 is implemented in much the same way. The main difference is that the error Hessian H^e replaces the Hessian H, and we find the dominant eigenpair of each of a sequence of eigenvalue problems, rather than finding p eigenpairs of the single Hessian H. Each iteration of a Lanczos-type solver for the eigenvalue problem in Algorithm 1 resembles that of Algorithm 2, and therefore the costs per iteration are asymptotically the same. It is more difficult to characterize the number of greedy iterations, and hence the number of eigenvector problems, that will be required using Algorithm 1. However, to the extent that the assumptions outlined in Section 3.1 hold, the number of greedy iterations will correspond roughly to the number of dominant eigenvalues of the full Hessian matrix H. As reasoned above, the spectrum of H is expected to decay rapidly for the problems of interest here; thus, convergence of the greedy reduced basis construction algorithm is expected to be rapid.

Chapter 4

Application: Convection-Diffusion Transport Problem

In this chapter, the model reduction methodology described in Chapter 3 is assessed for a contaminant transport problem. We first consider in Section 4.1 the case of a simple two-dimensional domain, which leads to a system of the form (2.8) of moderate dimension; in Section 4.2 a large-scale three-dimensional example will be presented.

4.1 Two-dimensional model problem

4.1.1 Problem description

The physical process is modeled by the convection-diffusion equation,

$$\frac{\partial w}{\partial t} + \vec{v} \cdot \nabla w - \kappa \nabla^2 w = 0 \quad \text{in } \Omega \times (0, t_f), \tag{4.1}$$

$$w = 0 \quad \text{on } \Gamma_D \times (0, t_f), \tag{4.2}$$

$$\frac{\partial w}{\partial n} = 0 \quad \text{on } \Gamma_N \times (0, t_f), \tag{4.3}$$

$$w = w_0 \quad \text{in } \Omega \text{ for } t = 0, \tag{4.4}$$

where w is the contaminant concentration (which varies in time and over the domain Ω), $\vec{v}$ is the velocity vector field, κ is the diffusivity, t_f is the time horizon of interest,

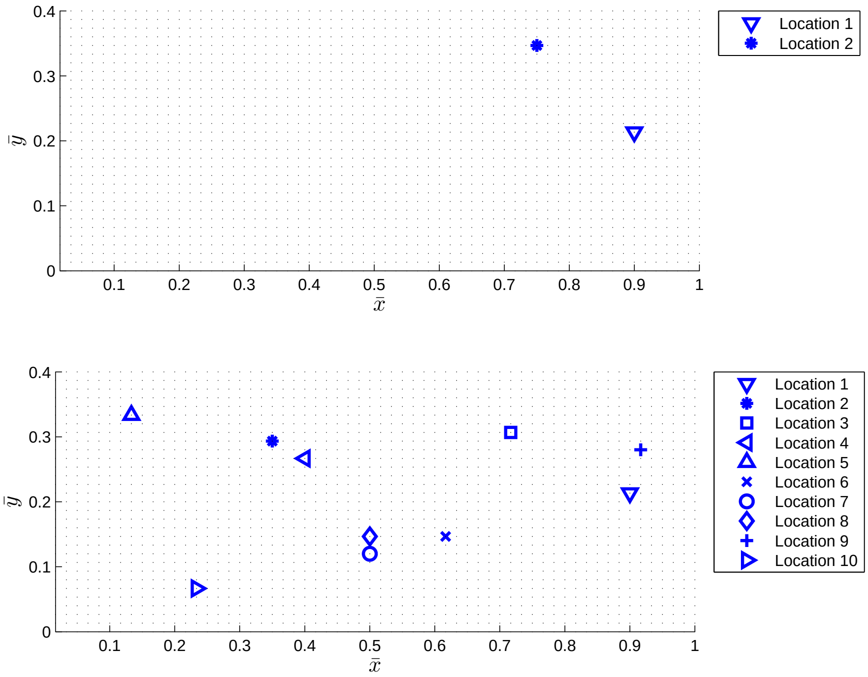


Figure 4-1: The computational domain and locations of sensor output nodes. Top: two-sensor case, bottom: ten-sensor case.

and w_0 is the given initial condition. Homogeneous Dirichlet boundary conditions are applied on the inflow boundary Γ_D, while homogeneous Neumann conditions are applied on the other boundaries Γ_N.

Figure 4-1 shows the computational domain for the two-dimensional contaminant transport example. The velocity field is taken to be uniform, constant in time, and directed in the positive $\bar{x}$-direction as defined by Figure 4-1. The inflow boundary, Γ_D, is defined by $\bar{x} = 0$, $0 \leq \bar{y} \leq 0.4$; the remaining boundaries comprise Γ_N.

A Streamline Upwind Petrov-Galerkin (SUPG) [10] finite-element method is employed to discretize (4.1) in space using triangular elements. For the cases considered here, the spatial mesh has $N = 1860$ nodes. The Crank-Nicolson method is used to discretize the equations in time. This leads to a linear discrete-time system of the form (2.8), where the state vector $x(k) \in \mathbb{R}^{1860}$ contains the values of contaminant concentration at spatial grid points at time t_k. For all experiments, the timestep used was $\Delta t = 0.02$ and the time limit, set approximately by the maximum time of convection across the length of the domain, was $t_f = 1.4$.

The matrix $\mathbf{A}$ in (2.8) depends on the velocity field and the Peclet number, Pe, which is defined as

$$\mathrm{Pe} = \frac{v_c \ell_c}{\kappa}, \tag{4.5}$$

where the characteristic velocity v_c is taken to be the maximum velocity magnitude in the domain, while the domain length is used as the characteristic length ℓ_c. The uniform velocity field described above was used in all experiments, but Pe was varied. Increasingly convective transport scenarios corresponding to Peclet numbers of 10, 100, and 1000 were used to generate different full-scale systems.

The outputs of interest are defined to be the values of concentration at selected sensor locations in the computational domain. Figure 4-1 shows two different sensor configurations that were employed in the results presented here.

The first step in creating a reduced model with Algorithm 2 is to compute p dominant eigenvectors of the full-scale Hessian matrix H. Figure 4-2 shows the eigenvalue spectra of H for the two-sensor case and the ten-sensor case. The relative decay rates of these eigenvalues are used to determine p, the number of eigenvectors used as seed initial conditions. We specify the parameter $\bar{\lambda}$, and apply the criterion that the jth eigenvector of H is included if $\lambda_j/\lambda_1 > \bar{\lambda}$.

Figure 4-2 demonstrates that the decay rate of the dominant eigenvalues is related to the number and positioning of output sensors. For the two-output case, the two dominant eigenvalues λ_1 and λ_2 are of almost equal magnitude; analogous behavior can be seen for the first ten eigenvalues in the ten-output case. This is consistent with the physical intuition that similarly important modes exist for each of the output sensors. For instance, a mode with initial concentration localized around one particular sensor is of similar importance as another mode with high concentration near a different sensor.

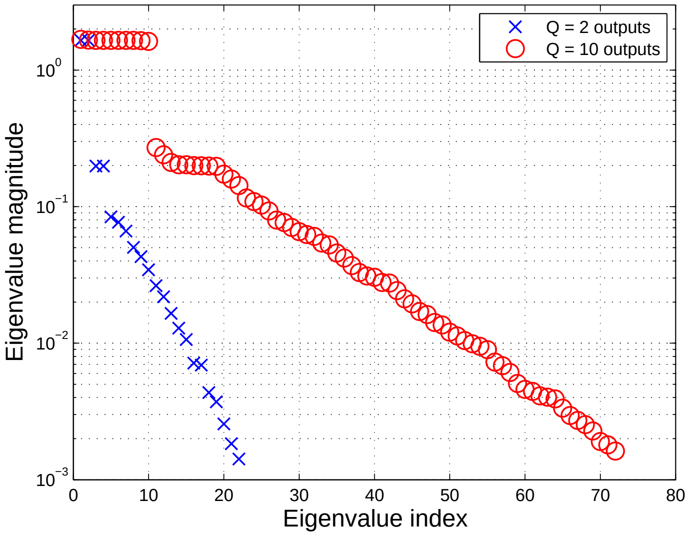


Figure 4-2: A comparison of the Hessian eigenvalue spectra of H for the two- and ten-output cases. Pe = 100.

4.1.2 Reduced model performance

Once the p seed eigenvectors have been computed, the corresponding state solutions, $\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^p$, are computed from (2.8) using each eigenvector in turn as the initial condition x_0. The final step in Algorithm 2 requires the formation of the reduced basis from the span of $\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^p$. We achieve this by aggregating all state solutions $x^i(k),\ i = 1, 2 \ldots, p,\ k = 0, 1, \ldots, T$ into a snapshot matrix $X \in \mathbb{R}^{N \times (T+1)p}$ and using the classical (non-weighted) POD to select the n basis vectors that most efficiently span the column space of X. The number of POD basis vectors is chosen based on the decay of the POD eigenvalues $\mu_1 \geq \mu_2 \geq \cdots \geq \mu_{(T+1)p} \geq 0$. As above, we define a parameter $\bar{\mu}$, and apply the criterion that the kth POD basis vector is retained if $\mu_k/\mu_1 > \bar{\mu}$.

The resulting reduced models given by (2.12), (2.13) can be used for any initial condition x_0; to demonstrate the methodology we choose to show results for initial conditions comprising a superposition of Gaussian functions. Each Gaussian is defined by

$$x_0(\bar{x}, \bar{y}) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\left[(\bar{x}-\bar{x}_c)^2 + (\bar{y}-\bar{y}_c)^2\right]/2\sigma^2}, \tag{4.6}$$

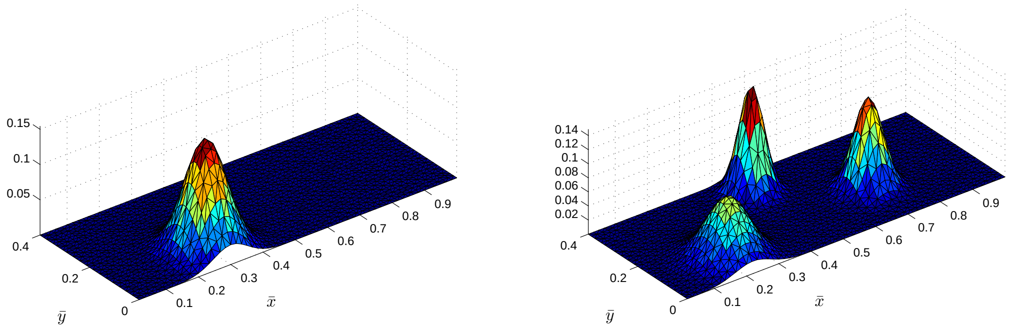
Case	$\bar{\lambda}$	$\bar{\mu}$	n	ε	ε_{max}
1	.1	10^{-4}	62	.0732	.1868
2	.1	10^{-6}	90	.0716	.1843
3	.01	10^{-4}	128	.0031	.0424
4	.01	10^{-6}	200	.0014	.0297
5	.001	10^{-4}	180	.00017	.0102
6	.001	10^{-6}	282	.00014	.0059

Table 4.1: Properties of various reduced-order models of a full-scale system with Pe=100 and two output sensors. The errors ε and ε_{max} are defined in (3.20) and (3.21), respectively; ε is evaluated when each reduced system (of dimension n) is subjected to test initial condition (c).

where $(\bar{x}_c, \bar{y}_c)$ defines the center of the Gaussian and σ is the standard deviation. All test initial conditions are normalized such that $||x_0||_2 = 1$. Three sample initial condition functions that are used in the following analyses are shown in Figure 4-3 and are referred to by their provided labels (a), (b), and (c) throughout.

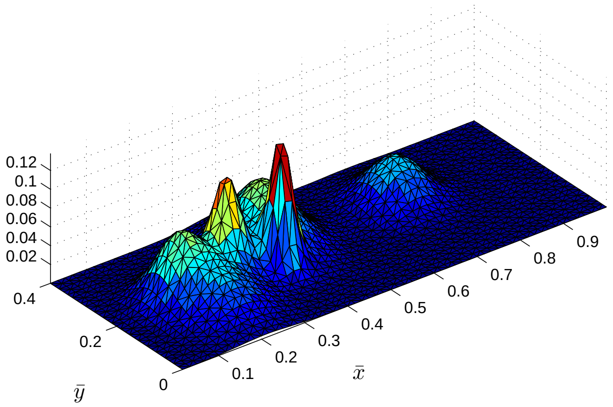
Tables 4.1 and 4.2 show sample reduced model results for various cases using the two-sensor configuration shown in Figure 4-1. The error ε is defined in (3.20) and computed for one of the sample initial conditions shown in Figure 4-3. It can be seen from the tables that a substantial reduction in the number of states from $N = 1860$ can be achieved with low levels of error in the concentration prediction at the sensor locations. The tables also show that including more modes in the reduced model, either by decreasing the Hessian eigenvalue decay tolerance $\bar{\lambda}$ or by decreasing the POD eigenvalue decay tolerance $\bar{\mu}$, leads to a reduction in the output error. Furthermore, the worst case error in each case, ε_{max}, is computed from (3.21) using the maximal eigenvalue of the error Hessian, H^e. It can also be seen that inclusion of more modes in the reduced model leads to a reduction in the worst-case error.

Figure 4-4 shows a comparison between reduced models computed using Algorithm 1 and Algorithm 2. The model sizes n increase with the number p of eigenvectors of either H or H^e used as seed initial conditions. For both algorithms, the maximum error ε_{max} and the error resulting from a forward solve with initial condition (b) decrease as p increases. The most dominant eigenvectors are similar: z_1 is



(a) Single Gaussian.

(b) Superposition of 3 Gaussians.



(c) Superposition of 7 Gaussians.

Figure 4-3: Sample test initial conditions used to compare reduced model outputs to full-scale outputs.

Case	$\bar{\lambda}$	$\bar{\mu}$	n	ε	ε_{max}
1	.1	10^{-4}	28	.0344	.2572
2	.1	10^{-6}	45	.0248	.2483
3	.01	10^{-4}	43	.0072	.0457
4	.01	10^{-6}	69	.0041	.0409
5	.001	10^{-4}	79	.00062	.0098
6	.001	10^{-6}	122	.00059	.0075

Table 4.2: Properties of various reduced-order models of a full-scale system with Pe=1000 and two output sensors. The errors ε and ε_{max} are defined in (3.20) and (3.21), respectively; ε is evaluated when each reduced system (of dimension n) is subjected to test initial condition (a).

identical to z_1^e, and $z_2 \approx z_2^e$. As p becomes large, though, the p^{th} eigenvector of H and the dominant eigenvector of H^e on the p^{th} iteration become increasingly different. This is evident when z_5 and z_5^e are plotted, since both are similar in shape but markedly different in their finer features. Despite this divergence, it can be seen that models formed using the one-shot method provide a similar level of accuracy as do the models formed with the iterative method, for the same reduced basis size n.

A representative comparison of full and reduced outputs, created by driving both the full and reduced systems with test initial condition (b), is shown in Figure 4-5 for the case of Pe=1000. The values $\bar{\lambda} = 0.01$ and $\bar{\mu} = 10^{-4}$ are used, leading to a reduced model of size $n = 196$. The figure demonstrates that a reduced model of size $n = 196$ formed using Algorithm 2 can effectively replicate the outputs of the full-scale system for this initial condition. The error for this case as defined in (3.20) is $\varepsilon = 0.0039$.

In order to ensure that the results shown in Figure 4-5 are representative, one thousand initial conditions are constructed randomly and tested using this reduced model. Each initial condition consists of 10 superposed Gaussian functions with random centers $(\bar{x}_c, \bar{y}_c)$ and random standard deviations σ. This library of test initial conditions was used to generate output comparisons between the full-scale model and the reduced-order model. The averaged error across all 1000 trials, $\bar{\varepsilon} = 0.0024$, is close to the error associated with the comparison shown in Figure 4-5. Furthermore,

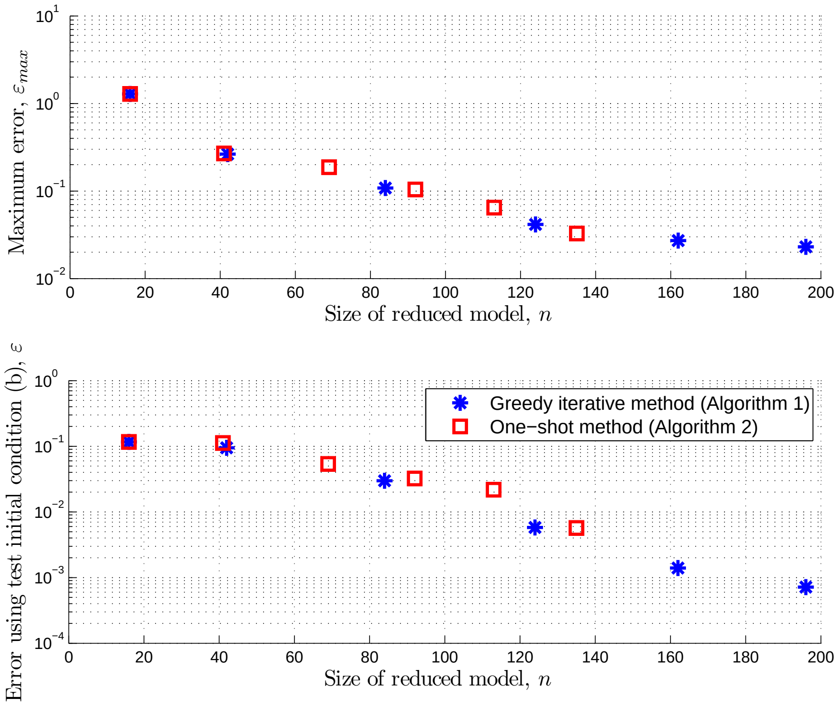


Figure 4-4: Top: Maximum error, ε_{max}, for reduced models computed using Algorithms 1 and 2. Bottom: Error for test initial condition (b), ε.

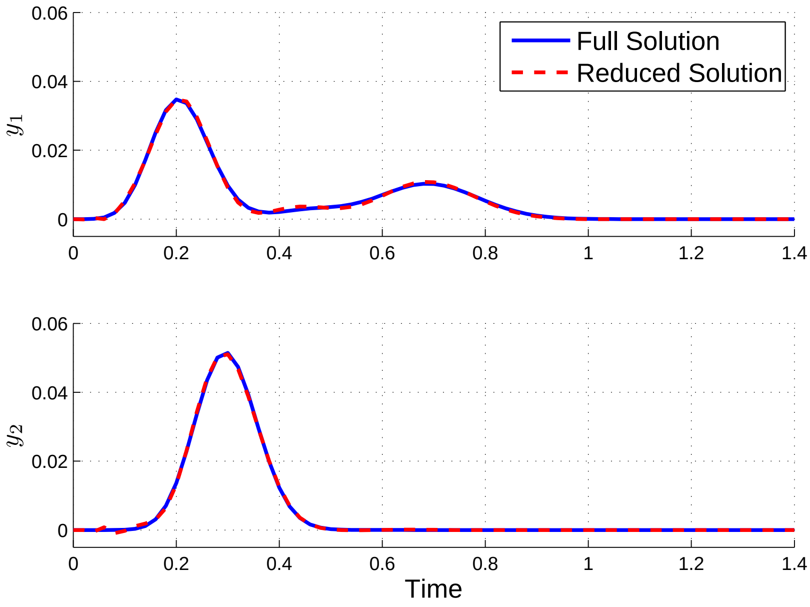


Figure 4-5: A comparison of full ($N = 1860$) and reduced ($n = 196$) outputs for two-sensor case using test initial condition (b). Pe=1000, $\bar{\lambda} = 0.01$, $\bar{\mu} = 10^{-4}$, $\varepsilon = 0.0039$.

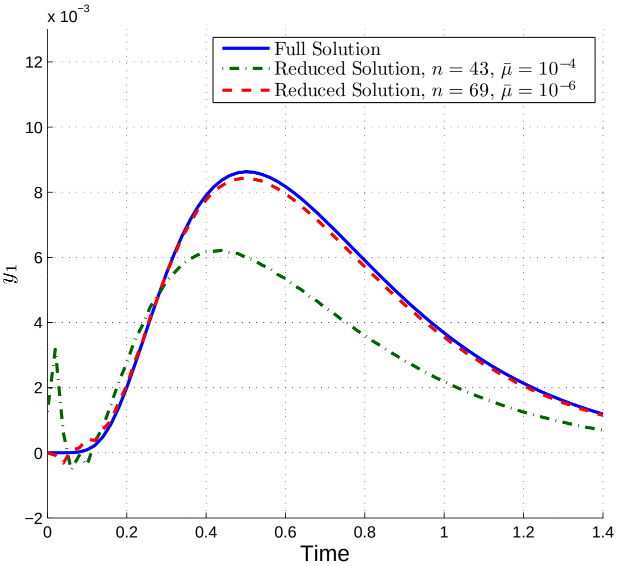


Figure 4-6: A comparison between full and reduced solutions at sensor location 1 for two different values of $\bar{\mu}$. Test initial condition (a) was used to generate the data. Pe=10, $\bar{\lambda} = 0.01$, two-sensor case. The output at the first sensor location is plotted here.

the maximum error over all 1000 trials is found to be 0.0059, which is well below the upper bound $\varepsilon_{max} = 0.0457$ established by (3.21).

Effect of variations in $\bar{\mu}$. As discussed above, $\bar{\mu}$ is the parameter that controls the number of POD vectors n chosen for inclusion in the reduced basis. If $\bar{\mu}$ is too large, the reduced basis will not span the space of all initial conditions for which it is desired that the reduced model be valid. Figure 4-6 illustrates the effect of changing $\bar{\mu}$. The curve corresponding to a value of $\bar{\mu} = 10^{-6}$ shows a clear improvement over the $\bar{\mu} = 10^{-4}$ case. This can also be seen by comparing the errors $\varepsilon = 0.0229$ and 0.0023 associated with the two reduced models seen in Figure 4-6. However, the improvement comes at a price, since the number of basis vectors, and therefore the size of the reduced model n, increases from 43 to 69 when $\bar{\mu}$ is decreased.

Effect of variations in $\bar{\lambda}$. Another way to alter the size and quality of the reduced model is to indirectly change p, the number of eigenvectors of H that are used as seed initial conditions for basis creation. We accomplish this by choosing different values of the eigenvalue decay ratio $\bar{\lambda}$. The effect of doing so is illustrated in Figure 4-7. An

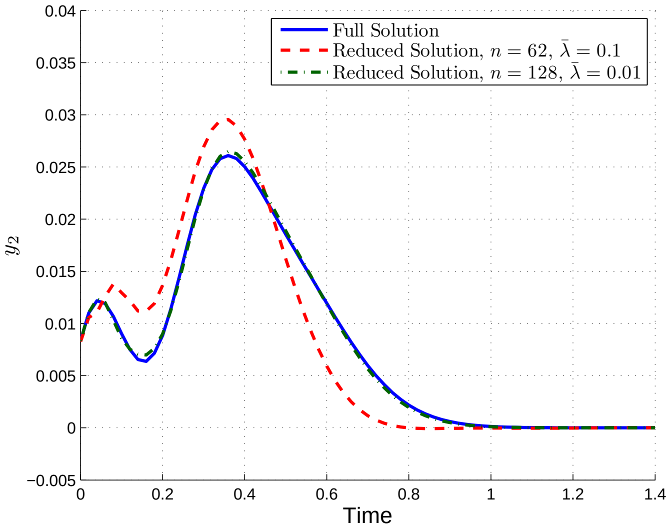


Figure 4-7: Lowering $\bar{\lambda}$ to increase p, the number of Hessian eigenvector initial conditions used in basis formation, leads to more accurate reduced-order output. Test initial condition (c) was used with two output sensors, Pe=100 and $\bar{\mu} = 10^{-4}$. The output at the second sensor location is plotted here.

increase in reduced model quality clearly accompanies a decrease in $\bar{\lambda}$. This can also be seen by comparing rows 1 and 3 of Table 4.1, which correspond to the two reduced models seen in Figure 4-7. The increase in n with lower values of $\bar{\lambda}$ is expected, since greater p implies more snapshot data with which to build the reduced basis, effectively uncovering more full system modes and decreasing the relative importance of the most dominant POD vectors. In general, for the same value of $\bar{\mu}$, more POD vectors are included in the basis if $\bar{\lambda}$ is reduced.

4.1.3 Ten-sensor case

To understand how the proposed method scales with the number of outputs in the system, we repeat the experiments for systems with $Q = 10$ outputs corresponding to sensors in the randomly-generated locations shown in Figure 4-1. A reduced model was created for the case of Pe=100, with $\bar{\mu} = 10^{-4}$ and $\bar{\lambda} = 0.1$. The result was a reduced system of size $n = 245$, which was able to effectively replicate all ten outputs of the full system. Figure 4-8 shows a representative result of the full and reduced

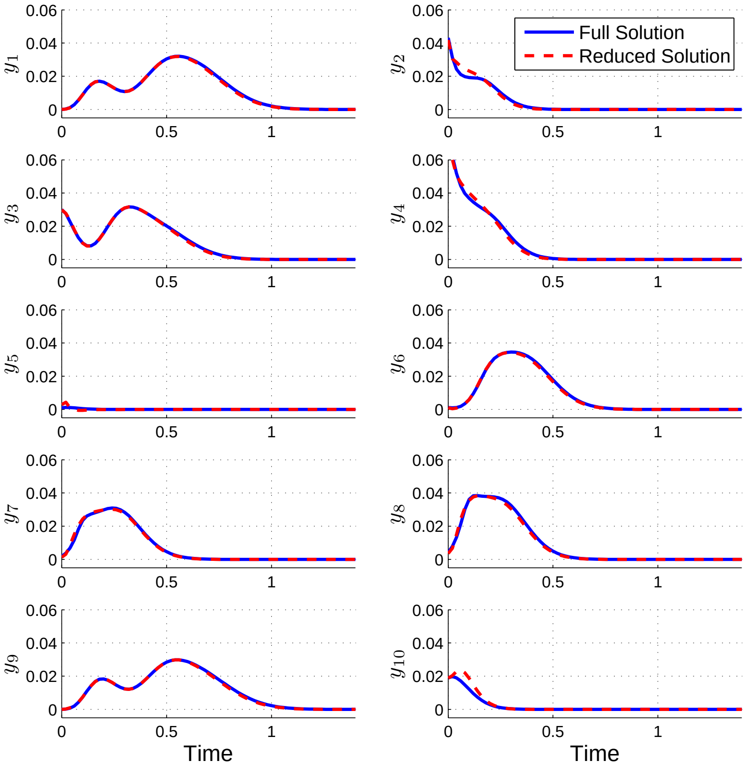


Figure 4-8: A comparison of the full ($N = 1860$) and reduced ($n = 245$) outputs for all $Q = 10$ locations of interest. Test initial condition (c) was used to generate these data with Pe = 100, $\bar{\mu} = 10^{-4}$, $\bar{\lambda} = 0.1$.

model predictions at all ten sensor locations.

The size $n = 245$ of the reduced model in this case is considerably larger than that in the corresponding two-output case ($n = 62$), which is shown in the first row of Table 4.1, although both models were constructed with identical values of $\bar{\mu}$ and $\bar{\lambda}$. The difference between high- and low-Q experiments is related to the Hessian eigenvalue spectrum. As demonstrated in Figure 4-2, the eigenvalue decay rate of the $Q = 10$ case is less rapid than that of the $Q = 2$ case. This means that, for the same value of $\bar{\lambda}$, more seed initial conditions are generally required for systems with more outputs. Since additional modes of the full system must be captured by the reduced model if the number of sensors is increased, it is not surprising that the size of the reduced basis increases.

4.1.4 Observations and Recommendations

The results above for the 2-D model problem demonstrate that reduced models formed by the proposed methods can be effective in replicating full-scale output quantities of interest. Algorithm 2 has also been shown to produce models of similar quality and size as models generated by Algorithm 1. Given also its straightforward implementation and lower offline cost (since we do not need to form the reduced model at each greedy iteration), Algorithm 2 is generally preferred for the construction of reduced bases. At this point, we can use the results to make recommendations about choosing $\bar{\mu}$ and $\bar{\lambda}$, the two parameters that control reduced-model construction.

In practice, one would like to choose these parameters such that both the reduced model size n and the modeling error for a variety of test initial conditions are minimal. The size of the reduced model is important because n is directly related to the online computational cost; that is, n determines the time needed to compute reduced output approximations, which is required to be minimal for real-time applications. The offline cost of forming the reduced model is also a function of $\bar{\mu}$ and $\bar{\lambda}$. When $\bar{\mu}$ is decreased, the basis formation algorithm requires more POD basis vectors to be computed; thus, decreasing μ increases the offline cost of model construction. In addition, the online cost of solving the reduced system in (2.12) and (2.13), which is not sparse, scales with n^2T. While decreasing $\bar{\mu}$ might appreciably improve modeling accuracy, doing so can only increase the time needed to compute reduced output approximations. Changes in $\bar{\lambda}$ affect the offline cost more strongly. Every additional eigenvector of H to be calculated adds the cost of several additional large-scale system solves: several forward and adjoint solves are needed to find an eigenvector using the matrix-free Lanczos solver described earlier. In addition, the number of columns of the POD snapshot matrix X grows by $(T+1)$ if p is incremented by one; computing the POD basis thus becomes more expensive. If these increases in offline cost can be tolerated, though, the results suggest a clear improvement in reduced-model accuracy for a relatively small increase in online cost.

Figure 4-9 illustrates the dependence of reduced model size and quality on the

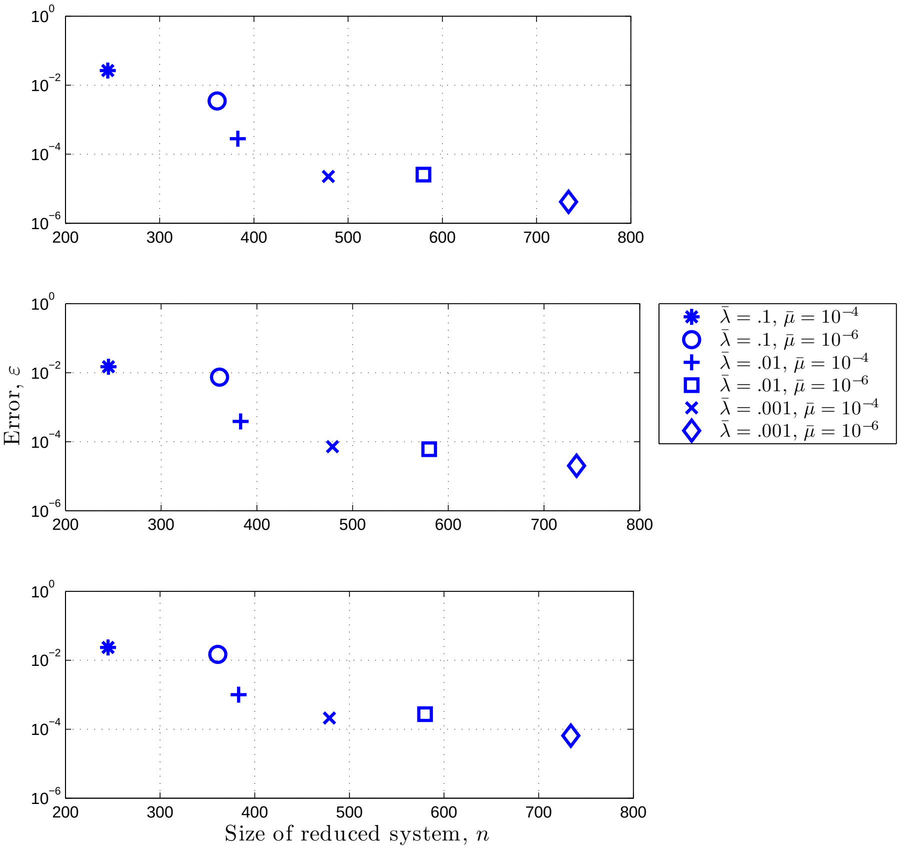


Figure 4-9: A measure of the error in six different reduced models of the same system plotted versus their sizes n for the ten-sensor case. The three plots were generated with test initial conditions (a), (b), and (c), respectively. Pe=100, $Q = 10$ outputs.

parameters $\bar{\mu}$ and $\bar{\lambda}$. For the case of ten output sensors with Pe=100, six different reduced models were constructed with different combinations of $\bar{\mu}$ and $\bar{\lambda}$. The three plots in Figure 4-9 show the error ε versus the reduced-model size n for each of the test initial conditions in Figure 4-3. Ideally, a reduced model should have both small error and small n, so we prefer those models whose points reside closest to the origin. Ignoring differences in offline model construction cost, decreasing $\bar{\lambda}$ should be favored over decreasing $\bar{\mu}$ if more accuracy is desired. This conclusion is reached by realizing that for a comparable level of error, reduced models constructed with lower values of $\bar{\lambda}$ are much smaller. Maintaining a small size of the reduced model is important for achieving real-time computations for large-scale problems of practical interest.

4.1.5 Output-Weighted POD Results

To this point, this chapter has presented results computed with the classical POD described in Section 2.2. In Section 2.3, though, we proposed a method to create reduced models by weighting snapshots before applying POD. These weights are dependent on the integral (2.27) of the output norm over time, beginning from the instant at which each snapshot is taken. The intended result of the output-weighted POD is the construction of reduced bases that require fewer basis vectors than do classical POD bases to provide the same degree of output accuracy. Alternatively, a basis formed with the weighted variant should provide greater accuracy with the same number of basis vectors as a classical POD basis.

To test the weighted method, results were collected for the ten-sensor case with $Pe = 100$ and $\bar{\lambda} = 0.01$. The parameter $\bar{\mu}$ was adjusted so that two reduced models with 383 basis vectors were constructed: one via classical POD, and the other using output-weighted POD. When the tight error bounds are compared, the results show a slight advantage to using the output-weighted POD ($\varepsilon_{max} = 0.0241$) over the classical POD ($\varepsilon_{max} = 0.0253$). In addition, when both reduced models are subjected to the same 1000 test initial conditions constructed from 10 superposed Gaussian functions with random centers and standard deviations, the results show that output-weighted POD, $\bar{\varepsilon} = 3.4699 \times 10^{-4}$, leads to a lower average error than does classical POD, $\bar{\varepsilon} = 4.2230 \times 10^{-4}$.

These small differences in maximal and average error can be explained by the low Peclet number in the domain. Since the seed initial conditions or dominant eigenvectors of the Hessian are generally localized in concentration about the sensor locations, the snapshots taken several timesteps after time zero correspond to smoother states in which the peaks in concentration have diffused outward. This means that those snapshots considered unimportant by the output-weighted POD are associated with low state energy as well as low output energy.

Even in this low-Pe case, though, the weighting scheme provides an added benefit. It was found that disregarding those snapshots with weights $\omega_i < 10^{-4}$ before applying non-weighted POD to the remaining snapshots produced the same reduced basis as did the non-weighted POD method with all snapshots included. Out of a total 3337 snapshots, 1445 were deemed unimportant by the above criteria. Since the cost of POD scales with the number of state solutions in the snapshot set, the exclusion led to a reduction in POD computation time.

Note that since the output weighting de-emphasizes the least-squares state energy optimization of classical POD, the weighted approach may not be advisable in a setting which requires full-state approximation based on the reduced states. Thus, the output-weighted POD is not used to construct the models used in Section 5.2.

4.2 Contaminant Transport in a 3-D Urban Canyon

We demonstrate our model reduction method by applying it to a three-dimensional airborne contaminant transport problem for which a solution is needed in real time. Intentional or unintentional chemical, biological, and radiological (CBR) contamination events are important security concerns. In particular, if contamination occurs in or near a populated area, predictive tools are needed to rapidly and accurately forecast the contaminant spread to provide decision support for emergency response efforts. Urban areas are geometrically complex and require detailed spatial discretization to resolve the relevant flow and transport, making prediction in real-time difficult. Reduced-order models can play an important role in facilitating real-time turn-around, in particular on mobile workstations in the field. However, it is essential that these reduced models be accurate over a wide range of initial conditions, since in principle any of these initial conditions can be realized. Once a suitable reduced-order model has been generated, it can serve as a surrogate for the full model within an inversion/data assimilation framework to identify the initial conditions given sensor data. The solution of an inverse problem using a reduced model is studied in Chapter 5.

To illustrate the generation of a reduced-order model that is accurate for high-dimensional initial conditions, we consider a three-dimensional urban canyon geometry occupying a (dimensionless) 15 × 15 × 15 domain. Figure 4-10 shows the domain and buildings, along with the locations of six output nodes that represent sensor locations of interest, all placed at a height of 1.5. The model used is again the convection-diffusion equation, given by (4.1). The PDE is discretized in space using an SUPG finite element method with linear tetrahedra, while the Crank-Nicolson method is used to discretize in time. Homogeneous Dirichlet boundary conditions of the form (4.2) are specified for the concentration on the inflow boundary, $\bar{x} = 0$, and the ground, $\bar{z} = 0$. Homogeneous Neumann boundary conditions of the form (4.3) are specified for the concentration on all other boundaries.

The velocity field, $\vec{v}$, required in (4.1) is computed by solving the steady laminar incompressible Navier-Stokes equations, also discretized with SUPG-stabilized linear tetrahedra. No-slip conditions, i.e. $\vec{v} = 0$, are imposed on the building faces and the ground $\bar{z} = 0$ (thus there is no flow inside the buildings). The velocity at the inflow boundary $\bar{x} = 0$ is taken as known and specified in the normal direction as

$$v_x(z) = v_{\max} \left(\frac{z}{z_{\max}} \right)^{0.5},$$

with $v_{\max} = 3.0$ and $z_{\max} = 15$, and zero tangentially. On the outflow boundary $\bar{x} = 15$, a traction-free (Neumann) condition is applied. On all other boundaries ($\bar{y} = 0$, $\bar{y} = 15$, $\bar{z} = 15$), we impose a combination of no flow normal to the boundary and traction-free tangent to the boundary. The spatial mesh for the full-scale system contains 68,921 nodes and 64,000 tetrahedral elements. For both basis creation and testing, a final non-dimensional time $t_f = 20.0$ is used, and discretized over 200 timesteps. The Peclet number based on the maximum inflow velocity and domain dimension is Pe=900. The PETSc library [7, 6, 8] is used for all implementation.

Figure 4-11 illustrates a sample forward solution. The test initial condition used in this simulation, meant to represent the system state just after a contaminant release event, was constructed using a Gaussian function with a peak magnitude of 100 centered at a height of 1.5.

For comparison with the full system, a reduced model was constructed using

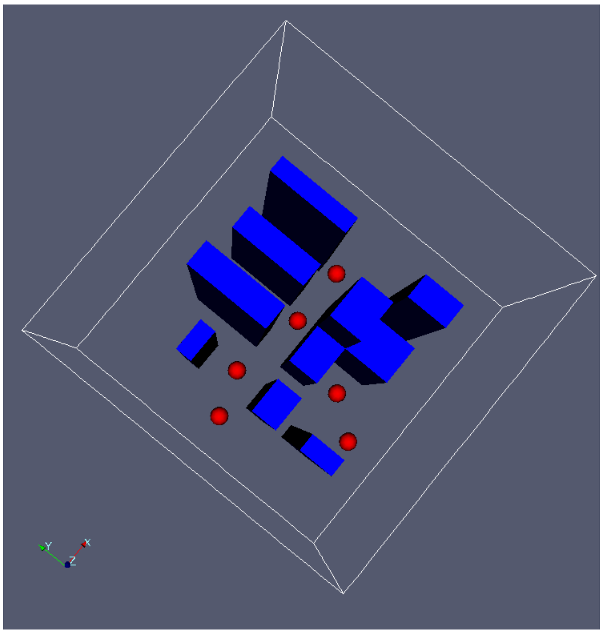


Figure 4-10: Building geometry and locations of outputs for the 3-D urban canyon problem.

Algorithm 2 with the eigenvalue decay ratios $\bar{\lambda} = 0.005$ and $\bar{\mu} = 10^{-5}$, which led to $p = 31$ eigenvector initial conditions and $n = 137$ reduced basis vectors. Eigenvectors were computed using the Arnoldi eigensolver within the SLEPc package [27], which is built on PETSc. Figure 4-12 shows a comparison of the full and reduced time history of concentration at each output location. The figure demonstrates that a reduced system of size $n = 137$, which is solved in a matter of seconds on a desktop, can accurately replicate the outputs of the full-scale system of size $N = 65,600$. We emphasize that the (offline) construction of the reduced-order model targets only the specified outputs, and otherwise has no knowledge of the initial conditions used in the test of Figure 4-12 (or any other initial conditions).

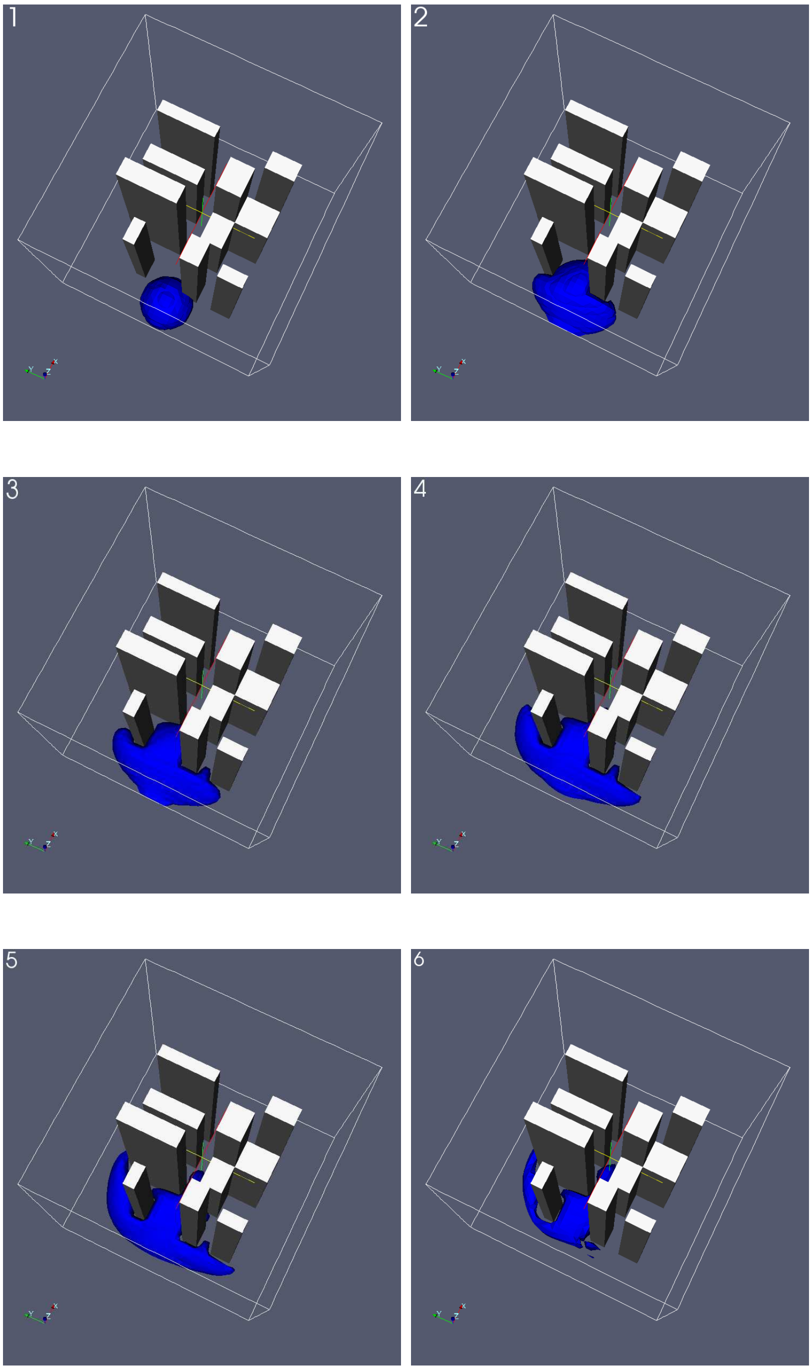


Figure 4-11: Transport of contaminant concentration through urban canyon at six different instants in time, beginning with the initial condition shown in upper left.

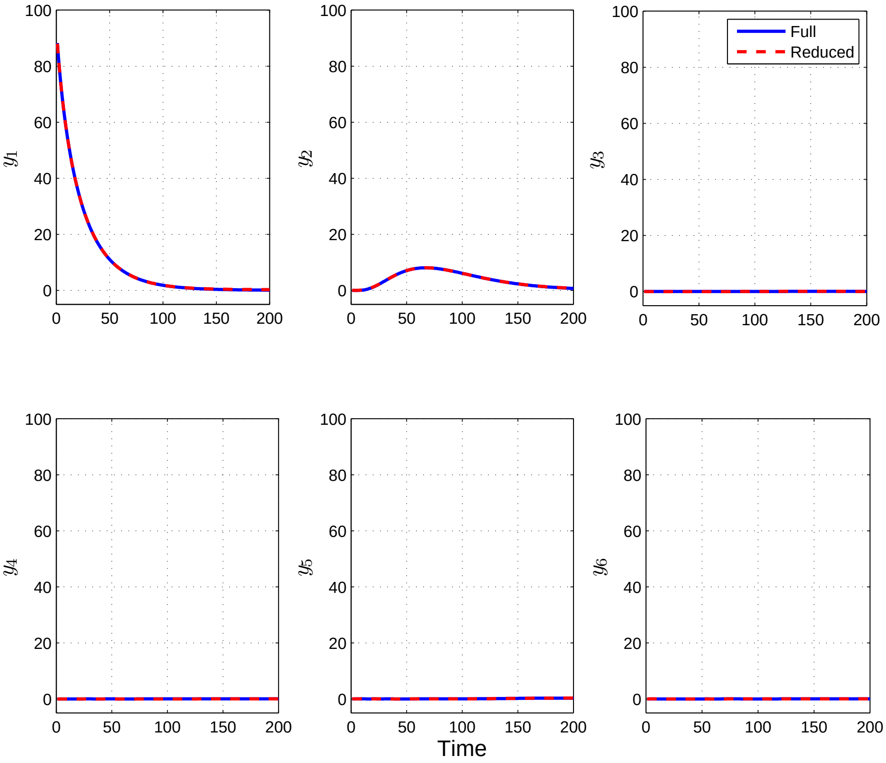


Figure 4-12: Full (65,600 states) and reduced (137 states) model contaminant concentration predictions at each of the six output nodes for the three-dimensional urban canyon example.

Chapter 5

Application: Measurement, Inversion, and Prediction

One intended application of the Hessian-based model-order reduction methodology is the efficient and accurate solution of time-dependent inverse problems. These problems involve estimating the initial condition of a system based on observations or measurements of system state later in time. For example, should a hazardous contaminant release occur inside a domain, the data provided by sensors that measure contaminant concentration can be used to reconstruct the initial distribution of material. Once this estimate of the initial condition is found, a prediction of state evolution can be issued: the future path of the contaminant can be computed based on the estimate.

The process of inversion based on measurements must take place in real time to be useful. Efficient methods of solution which rely on parallel computing are discussed in [3]; however, there has been no attempt to use reduced-order models as surrogates for the high-fidelity models that are typically used to represent system dynamics. Section 5.1 demonstrates that, since the Hessian-based methodology produces reduced models which can replicate full-scale outputs for a wide range of possible initial conditions, the reduced models are useful for solving inverse problems in real time.

The prediction of system state based on the estimate must also be performed in real time, since the contaminant path should be understood as soon as possible during

	Size	Description
x_0^a	N	Discrete representation of actual initial condition
x_0^t	N	Truth (full-order) initial condition estimate
x_0^{lr}	N	Low-rank initial condition estimate
x_{r0}^{rom}	n	Reduced-order initial condition estimate in reduced space
x_0^{rom}	N	Reduced-order initial condition estimate

Table 5.1: Summary of notation used to represent initial condition quantities.

a contaminant release event. The progression of full state can be found currently with a forward solve of the high-fidelity system of equations. In Section 5.2, we discuss the prospect of solving the reduced-order equations instead, using the reduced state at a given time to approximate the full state at that instant. This allows us to obtain approximately the same knowledge of state evolution, i.e. contaminant path, at diminished computational cost.

5.1 Estimating the Initial Condition

We wish to solve for estimates of the actual initial condition, defined as the system state at time $t = 0$, using various solution methods. Table 5.1 provides a summary of the notation we will use to represent each of the initial condition quantities.

To estimate the actual initial condition in the domain, we make use of observations such as sensor measurements from the initial time until the end of the data-collection time window. The window is divided into T_{obs} discrete timesteps, and the Q observations at each instant are stored in the vector $\mathbf{y}_{obs} \in \mathbb{R}^{Q(T_{obs}+1)}$, which has the same structure as shown in (2.10).

For simplicity, T_{obs} is chosen to be T, the maximum number of timesteps necessary for system state to come to equilibrium, i.e. for all contaminant to convect out of the domain after any initial distribution. In a situation requiring prediction, $T_{obs} < T$, and fewer timesteps are involved in the formation of the Hessian matrices and right-hand sides in (5.2), (5.5), and (5.10) below. This prediction based on a limited time horizon will be discussed in Section 5.3. However, the choice $T_{obs} = T$ suffices for demonstration of the inversion methods.

Subject to the governing equations, we search for an initial condition x_0^t which, when used to drive a forward solution of the system, produces outputs $\mathbf{y}$ that best match the observations $\mathbf{y}_{obs}$:

$$x_0^t = \arg\min \; (\mathbf{y} - \mathbf{y}_{obs})^T (\mathbf{y} - \mathbf{y}_{obs}) + \beta x_0^T x_0 \tag{5.1}$$

$$\begin{aligned} \text{where} \qquad \mathbf{A}\mathbf{x} &= \mathbf{F}x_0, \\ \mathbf{y} &= \mathbf{C}\mathbf{x}. \end{aligned}$$

Here, x_0^t represents the "truth" initial condition: the motivation for this nomenclature will become evident as approximate solutions to the inverse problem are explored. The constant β determines the relative weighting given to the regularization term $x_0^T x_0$. Regularization is required since, in general, the entire system state cannot be uniquely identified from sparse observations [39]. This means that the inverse problem is ill-posed: many initial conditions x_0 may lead to identical observations $\mathbf{y}_{obs}$. The regularization term, as written, is a filter which helps us select only smooth initial conditions by increasing the objective cost of states with sharp peaks and troughs. Thus, regularization is a means to selecting a single initial condition estimate out of many candidates which are consistent with observations.

The optimality conditions for (5.1) can be derived by first substituting the constraints into the objective function. The gradients of the objective function with respect to x_0 must be zero at the minimum, giving the following expression:

$$(H + \beta I)\, x_0^t = (\mathbf{C}\mathbf{A}^{-1}\mathbf{F})^T \mathbf{y}_{obs}, \tag{5.2}$$

which x_0^t must satisfy. The full-order or truth Hessian matrix $H = (\mathbf{C}\mathbf{A}^{-1}\mathbf{F})^T(\mathbf{C}\mathbf{A}^{-1}\mathbf{F})$ was introduced in Chapter 3. The ill-posedness of the inverse problem is related to the singularity of H, whose eigenvalue spectrum decays sharply to zero as shown in

Figure 4-2.

Reducing the computational cost associated with solving (5.2) is desirable: because the formation of $H \in \mathbb{R}^{N\times N}$ would require N forward and adjoint solves, neither H nor $(H+\beta I)^{-1}$ are formed explicitly. Matrix-free algorithms can be used to solve for x_0^t, but there are also ways to solve the inverse problem approximately. In the two sections that follow, we present two practical solution methods for the inverse problem described above. Section 5.1.4 compares the costs associated with implementing each solution method.

5.1.1 Low-Rank Hessian Approximation

A common approximate solution method involves forming a low-rank approximation [39] $H_{lr} \in \mathbb{R}^{N\times N}$ of the Hessian and using the Sherman-Morrison-Woodbury formula [20] to invert $H_{lr} + \beta I$. This method takes advantage of the compact eigenvalue spectrum of the Hessian [1].

The first step in computing $(H_{lr}+\beta I)^{-1}$ is the spectral decomposition of $H = U\Lambda U^{-1}$. We define $U_p \in \mathbb{R}^{N\times p}$ as a matrix whose columns are the p most dominant eigenvectors of H. Let $\Lambda_p \in \mathbb{R}^{p\times p}$ be a diagonal matrix of the p largest eigenvalues of H in descending order. The low-rank approximation of the Hessian can then be expressed as

$$H_{lr} = U_p \Lambda_p U_p^T. \tag{5.3}$$

Given that $(\beta I)^{-1} = \frac{1}{\beta} I$, the Sherman-Morrison-Woodbury formula helps us calculate $(H_{lr}+\beta I)^{-1}$ without the need to invert an $N \times N$ matrix:

$$(H_{lr}+\beta I)^{-1} = (U_p\Lambda_p U_p^T + \beta I)^{-1} = \frac{1}{\beta} I - \frac{1}{\beta} I U_p \left[\Lambda_p^{-1} + U_p^T \frac{1}{\beta} I U_p\right]^{-1} U_p^T \frac{1}{\beta} I. \tag{5.4}$$

It is now possible to compute an estimate of the initial condition which relies on the low-rank approximation of H. This estimate matches the truth solution x_0^t in the limiting case $p = N$. We will refer to any inverse problem solution estimate derived from a low-rank Hessian approximation as x_0^{lr}:

$$(H_{lr} + \beta I)x_0^{lr} = (\mathbf{C}\mathbf{A}^{-1}\mathbf{F})^T\mathbf{y}_{obs}. \tag{5.5}$$

5.1.2 Reduced-Order Inverse Problem Solution

The low-rank approximation described in the previous section is an attempt to reduce the full-scale Hessian after its construction from the high-fidelity system of equations. Conversely, we propose in this section a means to solve the inverse problem approximately by first reducing the high-fidelity system and then forming a reduced-order analogue of (5.2).

This solution method is initiated with the use of the Hessian-based approach described in Chapter 3 to construct a suitable reduced basis V. Recall that once this basis is obtained, the block components A_r and C_r of the reduced-order system of equations in (2.12)–(2.13) are defined. This allows us to write an optimization problem similar to (5.1) to estimate the initial condition $x_{r0} \in \mathbb{R}^n$ in *reduced* space:

$$x_{r0}^* = \arg\min \;\; (\mathbf{y}_r - \mathbf{y}_{obs})^T (\mathbf{y}_r - \mathbf{y}_{obs}) + \beta x_{r0}^T x_{r0} \tag{5.6}$$

$$\begin{aligned} \text{where} \qquad \mathbf{A}_r\mathbf{x}_r &= \mathbf{f}_r x_{r0}, && (5.7)\\ \mathbf{y}_r &= \mathbf{C}_r\mathbf{x}_r, && (5.8) \end{aligned}$$

and $\mathbf{f}_r \in \mathbb{R}^{n(T+1)\times n}$ has replaced F_r in the description of the reduced-order system. The matrix $\mathbf{f}_r$ contains E_r from (2.5) in its first $n \times n$ block and only zeros in the subsequent T blocks:

$$\mathbf{f}_r = \begin{bmatrix} E_r \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}. \tag{5.9}$$

As seen in the first constraint (5.7), the use of $\mathbf{f}_r$ in place of $\mathbf{F}_r$ means that the reduced-order state equations are driven from an initial condition in reduced-space: x_{r0} replaces x_0. This substitution is made so that the reduced-order Hessian, $H_r = (\mathbf{f}_r\mathbf{A}_r^{-1}\mathbf{C}_r)^T(\mathbf{f}_r\mathbf{A}_r^{-1}\mathbf{C}_r)$, has dimension $n \times n$ as opposed to the $N \times N$ matrix which would have resulted from the use of (2.12) and $\mathbf{F}_r$. The two formulations are equivalent since $x_{r0} = V^T x_0$.

Using the same reasoning as above to write a closed-form solution to the optimization problem, the analogue to (5.2) in the reduced-order model case is given by

$$(H_r + \beta I)\, x_{r0}^{rom} = (\mathbf{C}_r\mathbf{A}_r^{-1}\mathbf{f}_r)^T\mathbf{y}_{obs}, \tag{5.10}$$

where x_{r0}^{rom} is the estimated initial condition in reduced space. To compute the estimated initial condition x_0^{rom} in full-order space, we make use of the approximation (2.4):

$$x_0^{rom} = V x_{r0}^{rom}. \tag{5.11}$$

5.1.3 Inverse Problem Results

An experiment based on the 2-D contaminant transport problem described in Chapter 4 was devised to compare the three methods – truth solution, low-rank approximation, and reduced-order solution – of estimating the initial condition. Figure 5-1 illustrates the actual initial condition which must be identified.

Although the actual initial condition is defined on a grid with 7320 finite element nodes, the computations in this section are performed on a coarser grid with $N = 1860$ nodes. This choice is made in an effort to avoid the "inverse crime" as explained by [12]. In practice, the actual initial condition is independent of any spatial discretization and is never precisely known. In order to compare this actual initial condition to estimates computed based on the coarse grid, we define $x_0^a \in \mathbb{R}^{1860}$ as a vector composed of samples of the distribution in Figure 5-1 at the $(\bar{x}, \bar{y})$ locations of

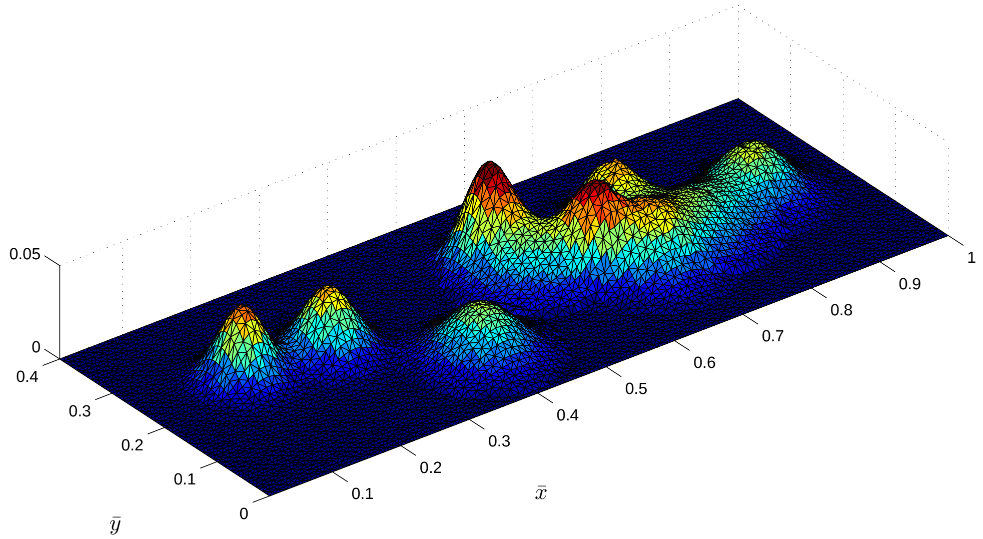


Figure 5-1: The actual initial condition used for the experiments; the goal of solving the inverse problem is to find this distribution from sensor measurements. Formed by superposing 10 Gaussian distributions with random centers and standard deviations.

the coarse grid nodes.

The domain contains $Q = 10$ sensors in the locations shown at the bottom of Figure 4-1. From the actual initial condition, the high-fidelity system of equations are solved forward in time on the fine grid for T_{obs} additional timesteps to provide the outputs $\mathbf{y} \in \mathbb{R}^{Q(T_{obs}+1)}$, or the concentrations over time at each of the sensor locations. However, these output values are not directly used as observed measurements. It is assumed that all sensor measurements are only accurate to within 5% of the true concentration at each sensor location. We introduce noise at a certain time t_i for a given sensor q by incrementing the actual value $y^q(t_i)$ by $\eta y^q(t_i)$, where η is chosen randomly with a uniform probability distribution between -0.05 and 0.05. Thus, $\mathbf{y}_{obs}$ is $\mathbf{y}$ with noise introduced to each element, and each sensor is assumed to exhibit the same potential for error at each timestep. The combination of using a fine grid and adding noise to the computed data provides a more realistic setting in which to test the methodology.

The number of timesteps T_{obs} with length $\Delta t = 0.02$ in the observation window is chosen such that the time limit, set approximately by the maximum time of convection across the length of the domain, is $t_f = 1.4$. This is a poor choice for a practical setting in which prediction is important, since the contaminant will have been transported throughout the domain by the time all measurements are taken; however, this choice ensures a fair comparison between the low-rank approximation and reduced-order inverse solution methods.

To form the reduced-order models necessary for solving the inverse problem, we use Algorithm 2. A model formed with parameters $\bar{\lambda} = 0.1$ and $\bar{\mu} = 10^{-4}$ and size $n = 245$ will be referred to as the baseline model. We also define "strict" parameters $\bar{\lambda} = 0.01$ and $\bar{\mu} = 10^{-6}$ which can be used in place of the baseline values to create varied reduced-order models.

With the pre-existing set of full-order state evolution equations in the constraints of (5.1) and the choice of T_{obs}, it is possible to construct the Hessian matrix $H \in \mathbb{R}^{1860\times1860}$ from (5.2). This matrix is necessary for both the truth solution and the low-rank approximation. We can also use the baseline parameters to construct the reduced-order Hessian $H_r \in \mathbb{R}^{245\times245}$ in (5.10) for use in finding the reduced estimate of initial condition. In practice, H and H_r are not formed explicitly: we require instead a function that provides the action of each matrix on a vector. However, the relatively small number of full-scale unknowns $N = 1860$ in our model problem makes their construction feasible. The partial eigenvalue spectra of $H+\beta I$ and $H_r+\beta I$, with and without regularization, are shown in Figure 5-2. The plot shows the eigenvalues of H decay slightly less sharply than those of H_r; with a regularization constant of $\beta = 0.001$, the spectra are similar. The choice of regularization constant for our experiment was made empirically by solving for the truth initial condition in (5.2) with different values of β. If the regularization constant is too small, both (5.2) and (5.10) have multiple solutions. In the other extreme, the estimated initial condition becomes infinitely smooth since the first terms in the objective functions of (5.1) and (5.6) are negligible. The ideal value of β is as small as possible while still leading to a unique solution.

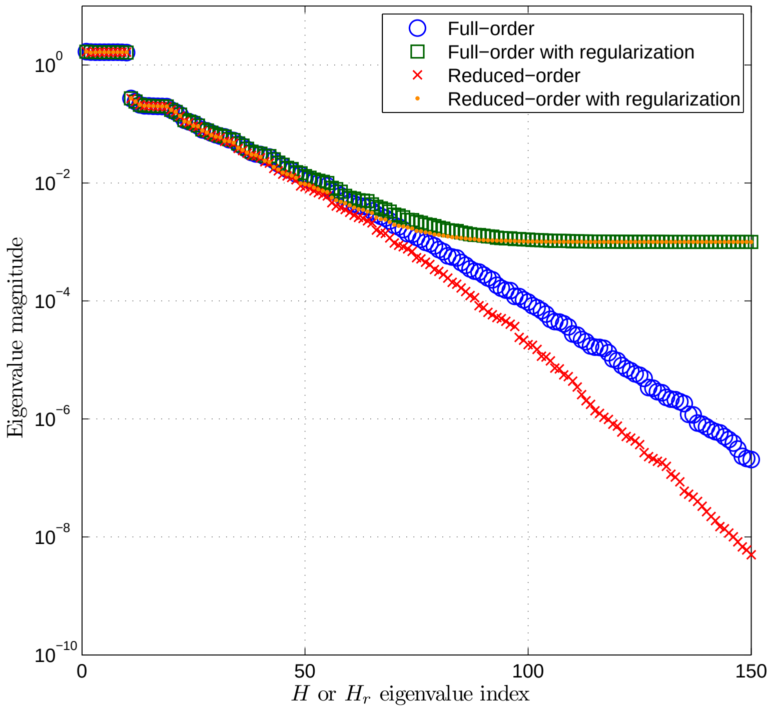


Figure 5-2: The first 150 eigenvalues in the spectra of $H + \beta I$ and $H_r + \beta I$ with $\beta = 0$ and $\beta = 0.001$. The reduced model used to construct H_r is the baseline model with $\bar{\lambda} = 0.1$, $\bar{\mu} = 10^{-4}$, and $n = 245$. $Pe = 100$, $Q = 10$ outputs.

The objective of the experiment is to compare the low-rank and reduced-order initial conditions to both the truth initial condition and the actual initial condition. Low-rank approximations computed with different numbers p of eigenpairs of H were used to generate multiple estimates of the initial condition x_0^{lr}. In addition, reduced-order models formed with different combinations of baseline and strict parameters were used to compute varying estimates x_0^{rom}. Recall that $\bar{\lambda}$ is the parameter which controls how many eigenvectors of H are required to form a given reduced model. We desire an accurate estimate of the initial condition which requires the fewest possible eigenpairs of H, the computation of which is the primary source of offline cost.

A summary of the results is shown in Figure 5-3. When compared to either the truth or the actual initial condition, the reduced-order estimates require fewer eigenvectors of H than do the low-rank estimates to achieve the same level of accuracy. Each low-rank approximation is of order p, while each reduced-order model is typically associated with its size n. Despite this difference, we compare all errors versus p since this quantity controls the offline cost in both cases.

On the bottom of the figure, the error with respect to the truth initial condition x_0^t is plotted. It can be seen that reduced-order models formed with 47 eigenvectors of H provide estimates which demonstrate the same accuracy as the low-rank estimate with 80 modes. Low-rank estimates made based on increasingly smaller values of p exhibit sharply increasing error. If the regularization constant is changed from $\beta = 0.001$ to $\beta = 0.01$, the error does not grow as sharply, but the low-rank estimates still demonstrate markedly larger error than do the reduced-order estimates. These results show that the process of computing snapshots and applying POD in the course of building a reduced model extracts more useful information from a given number of eigenvectors of H than does simply using the p eigenvectors to form the low-rank approximation.

The top of the figure shows the error with respect to the coarse grid representation x_0^a of the actual initial condition. Here, we note that, although the low-rank error is even greater than 1 for 50 eigenvectors of H or fewer, the errors associated with the reduced estimates formed with only 21 eigenvectors are close to the error 0.18 associ-

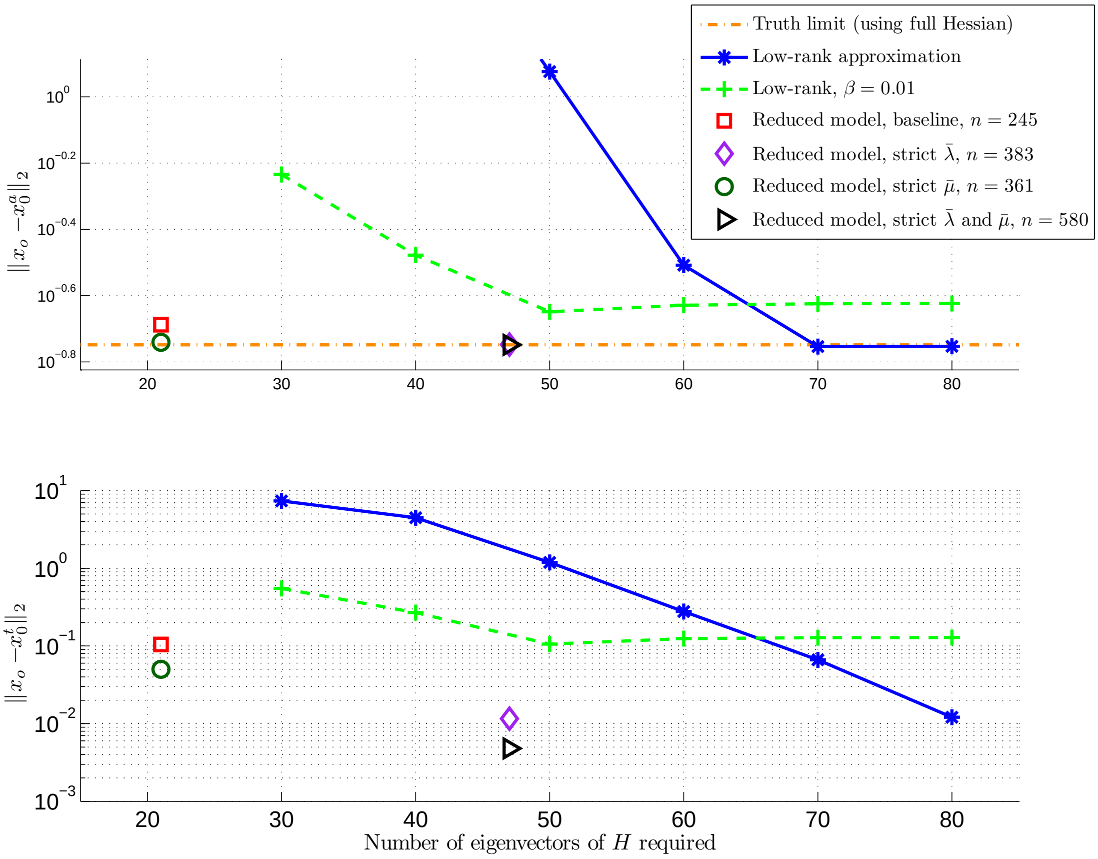
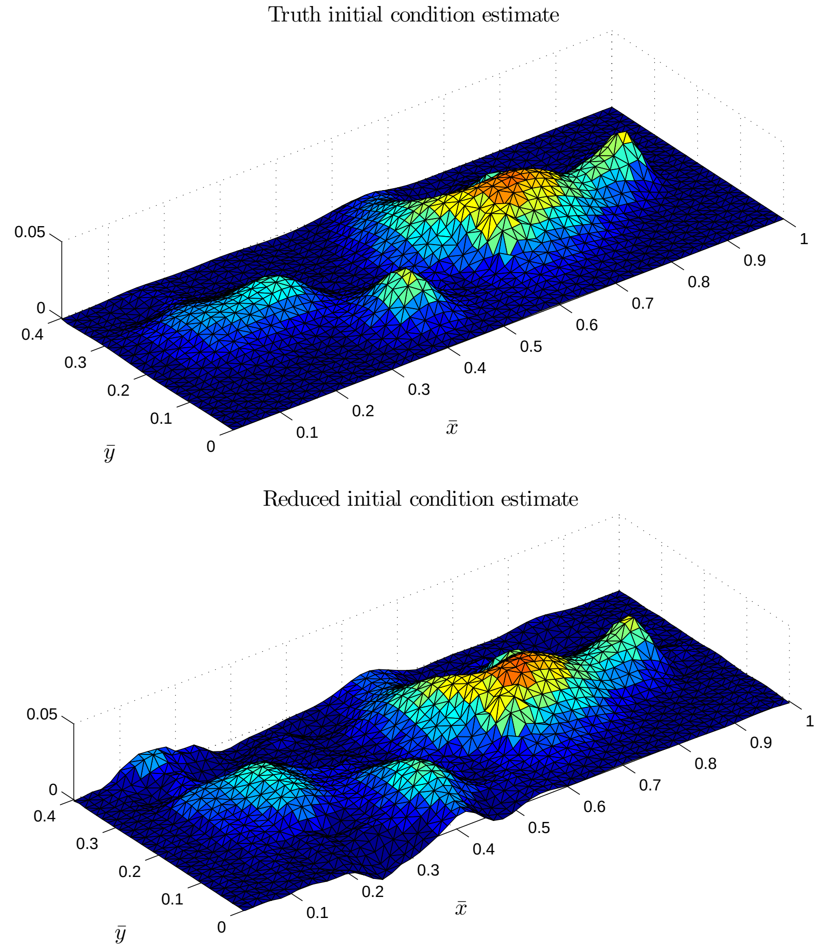


Figure 5-3: Error in low-rank and reduced-order initial condition versus actual initial condition x_0^a (top) and versus truth initial condition x_0^t (bottom). Data from reduced models formed with baseline values of $\bar{\lambda} = 0.1$ and $\bar{\mu} = 10^{-4}$ and with strict values of $\bar{\lambda} = 0.01$ and $\bar{\mu} = 10^{-6}$ are shown. Unless otherwise stated, the regularization constant for all trials is $\beta = 0.001$. $Pe = 100$, $Q = 10$ outputs.

ated with the truth initial condition. We conclude that the additional computational effort needed to form reduced-order models with strict $\bar{\lambda}$ may not be necessary: in this case, the models formed with few eigenvectors provide sufficiently accurate estimates of the actual initial condition.

Figure 5-4 illustrates both the truth initial condition and the baseline reduced-order initial condition, demonstrating that the prominent features of x_0^t are replicated in x_0^{rom}. Furthermore, Figure 5-5 shows the extent to which the reduced-order initial condition, when used to drive a forward solve of the reduced-order equations, yields outputs that match those associated with both the truth and actual cases.


Truth initial condition estimate

Reduced initial condition estimate

Figure 5-4: The truth initial condition estimate x_0^t (top) and the reduced-order estimate x_0^{rom}. The reduced model used is the baseline model with $\bar{\lambda} = 0.1$, $\bar{\mu} = 10^{-4}$, and $n = 245$. $Pe = 100$, $Q = 10$ outputs.

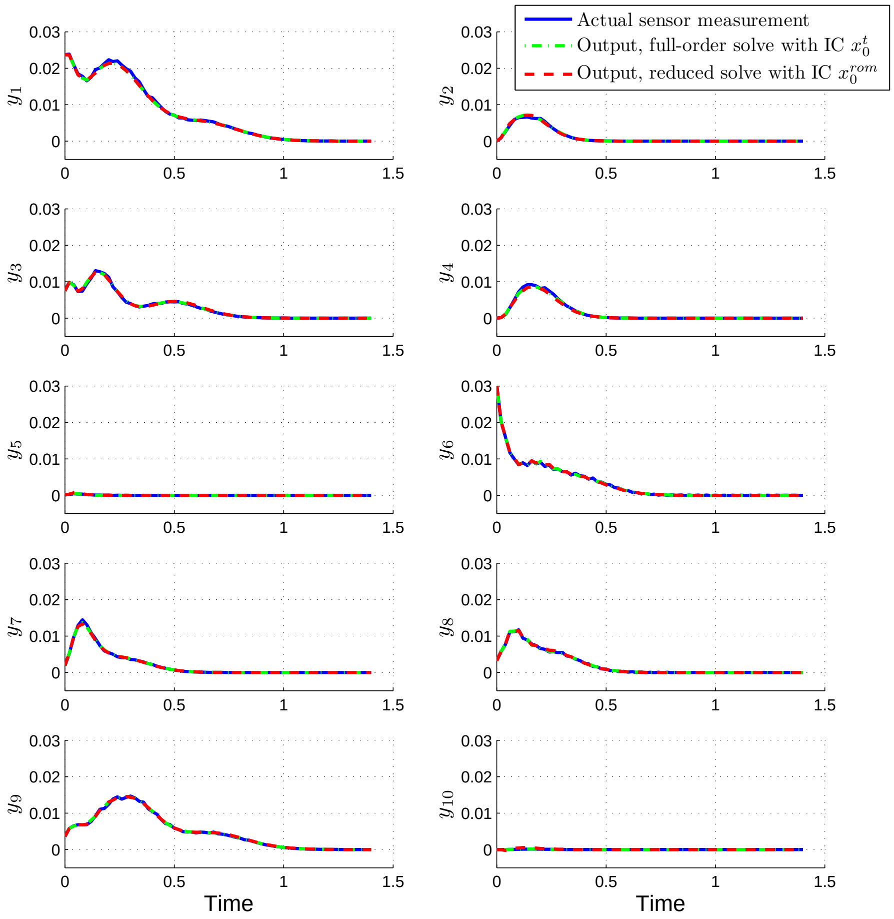


Figure 5-5: The the actual sensor measurements (outputs) compared to $\mathbf{y}$ and $\mathbf{y}_r$. Here, $\mathbf{y}$ is generated by solving forward in time with the full-order equations starting from truth initial condition x_0^t; $\mathbf{y}_r$ is generated by solving forward in time with the reduced-order equations starting from reduced initial condition x_0^{rom}. Both initial condition estimates are shown in Figure 5-4.

5.1.4 Implementation Cost Comparison

As discussed above, two practical choices for initial condition estimation are the low-rank approximation and reduced-order inversion. Both methods can be divided into offline and online (real-time) phases: we discuss the cost of each phase in this section, noting that the division between phases may differ in practice.

The terms $(H_{lr} + \beta I)^{-1}$ and $V(H_r + \beta I)^{-1}$ necessary for the solution of (5.5) and (5.10)–(5.11), respectively, can be computed *offline* if the number of timesteps in the observation window is chosen beforehand. We can take advantage of matrix-free iterative methods such as Lanczos to find offline the p eigenpairs of H that are required indirectly for both the low-rank and reduced-order inversion methods. The previous section cites this process as the primary source of offline cost in both cases.

The additional steps of the low-rank process which involve significant offline cost include the inversion of the $p \times p$ matrix $\Lambda_p^{-1} + U_p^T \frac{1}{\beta} I U_p$. The quantity $(H_{lr} + \beta I)^{-1}$ can then be formed via the series of matrix multiplications in (5.4). The reduced-order method involves additional offline effort as well. Assuming that Algorithm 2 is used to generate the reduced model, p forward solves of the high-fidelity equations are necessary to generate snapshots. However, since iterative eigensolvers employ repeated forward solutions, it may be possible to extract the required snapshots from the final iteration used to find the p eigenvectors of H.

A major step in the reduced-order process which is absent in the low-rank approximation is the POD: the singular value decomposition is applied to the $N \times p(T+1)$ snapshot matrix, where N is expected to be much greater than $p(T+1)$. (The full-order Hessian matrix is of order $N \times N$, so the computation of its eigenvectors still dominates the offline cost of the reduced-order method.) Computing $(H_r + \beta I)^{-1}$ involves n forward and adjoint solves and the inversion of an $n \times n$ matrix if H_r is formed explicitly. Finally, multiplying the basis matrix $V \in \mathbb{R}^{N \times n}$ by $(H_r + \beta I)^{-1} \in \mathbb{R}^{n \times n}$ is required to form $V(H_r + \beta I)^{-1}$.

The goal of the *online* phase is to compute x_0^{lr} or x_0^{rom} as quickly as possible in real time after the measurements $\mathbf{y}_{obs}$ become available. It involves solution of (5.5) given $(H_{lr}+\beta I)^{-1}$ or solution of (5.10)-(5.11) given $V(H_r+\beta I)^{-1}$. These low-rank and reduced-order inversion equations are rewritten below in a convenient form:

$$
\begin{aligned}
x_0^{lr} &= \underbrace{(H_{lr}+\beta I)^{-1}}_{N\times N}(\mathbf{C}\mathbf{A}^{-1}\mathbf{F})^T\mathbf{y}_{obs}, && (5.12)\\
x_0^{rom} &= \underbrace{V(H_r+\beta I)^{-1}}_{N\times n}(\mathbf{C}_r\mathbf{A}_r^{-1}\mathbf{f}_r)^T\mathbf{y}_{obs}, && (5.13)
\end{aligned}
$$

where the highlighted quantities are computed offline. The matrices $\mathbf{C}$, $\mathbf{A}$, $\mathbf{F}$, $\mathbf{C}_r$, $\mathbf{A}_r^{-1}$, and $\mathbf{f}_r$ are known before the real-time phase as well. The three full-order quantities are not explicitly formed since their sizes scale with multiples of NT; however, the reduced-order matrices, whose sizes scale with multiples of nT, can be computed and stored inexpensively.

The first task once $\mathbf{y}_{obs}$ is known is to compute $(\mathbf{C}\mathbf{A}^{-1}\mathbf{F})^T\mathbf{y}_{obs}$ in the low-rank case or $(\mathbf{C}_r\mathbf{A}_r^{-1}\mathbf{f}_r)^T\mathbf{y}_{obs}$ in the reduced-order case. Multiplication by $\mathbf{A}^{-T}$ is equivalent to performing an adjoint solve in the high-fidelity space, whereas multiplication by $\mathbf{A}_r^{-T}$ represents an adjoint solve in the reduced space. Consequently, performing this step with the reduced-order method is advantageous in terms of computational cost. The advantage is even greater if the quantity $(\mathbf{C}_r\mathbf{A}_r^{-1}\mathbf{f}_r)^T \in \mathbb{R}^{n\times Q(T+1)}$ is computed offline and stored.

The remaining online cost is associated with evaluating x_0^{lr} or x_0^{rom} from previously-computed quantities. This involves multiplying the matrix $(H_{lr}+\beta I)^{-1} \in \mathbb{R}^{N\times N}$ or $V(H_r+\beta I)^{-1} \in \mathbb{R}^{N\times n}$ by the vector $(\mathbf{C}\mathbf{A}^{-1}\mathbf{F})^T\mathbf{y}_{obs} \in \mathbb{R}^N$ or $(\mathbf{C}_r\mathbf{A}_r^{-1}\mathbf{f}_r)^T\mathbf{y}_{obs} \in \mathbb{R}^n$, respectively. The multiplication in the low-rank case requires $\mathcal{O}(N^2)$ work while its reduced-order counterpart requires $\mathcal{O}(Nn)$ work. The reduced-order method thus involves less computational effort in both parts of the online phase.

In addition, having a reduced model available is useful in the forward solve which many applications demand after the initial condition estimate is found. This added benefit is discussed in Section 5.2.2.

5.2 Using Reduced State to Approximate Full State

In Chapter 4, we demonstrated that reduced-order models constructed by the Hessian-based approach can replicate full-scale outputs at lower computational cost. By using only seed initial conditions which heavily influence the outputs of interest, the Hessian-based algorithms place more emphasis on output information than state information. It is still possible, though, to obtain an approximation $\hat{x}(k)$ of full state from the reduced-order state $x_r(k)$ at time instant k. We do this by applying (2.4):

$$\hat{x}(k) = V x_r(k).$$

5.2.1 Comparison of Full State to its Approximation

The full-state approximation (2.4) is useful in the forward solution of large-scale dynamical systems since the forward solve can be computed in reduced-space at lower computational cost. If full-state information, i.e. a snapshot of contaminant in our experimental domain, is required at a certain time instant, then one matrix-vector multiplication can be computed to yield the snapshot approximation.

To evaluate the quality of this approximation, the same test initial condition was used to drive both full and reduced systems, (2.8)–(2.9) and (2.12)–(2.13), respectively. Figure 5-6 shows the result when three different reduced models with ten outputs are used to solve forward in time, with a full-state approximation $V x_r$ obtained at each timestep. The plot illustrates that, while each error curve decays with time due to diffusion, the reduced model of size $n = 383$ formed with a strict value of $\bar{\lambda}$ maintains a smaller error norm over the time horizon than does the similarly sized model ($n = 361$) constructed with strict $\bar{\mu}$. This is consistent with the expectation that a more diverse set of seed initial conditions provides the reduced basis with more state information than does simply adding more basis vectors after POD is applied to an existing snapshot set. Thus, if more accurate full-state approximation is a priority, reduced-order models with strict $\bar{\lambda}$ should be considered. It should be noted that the choice of outputs, i.e. number of sensors and their locations, controls the eigenvectors

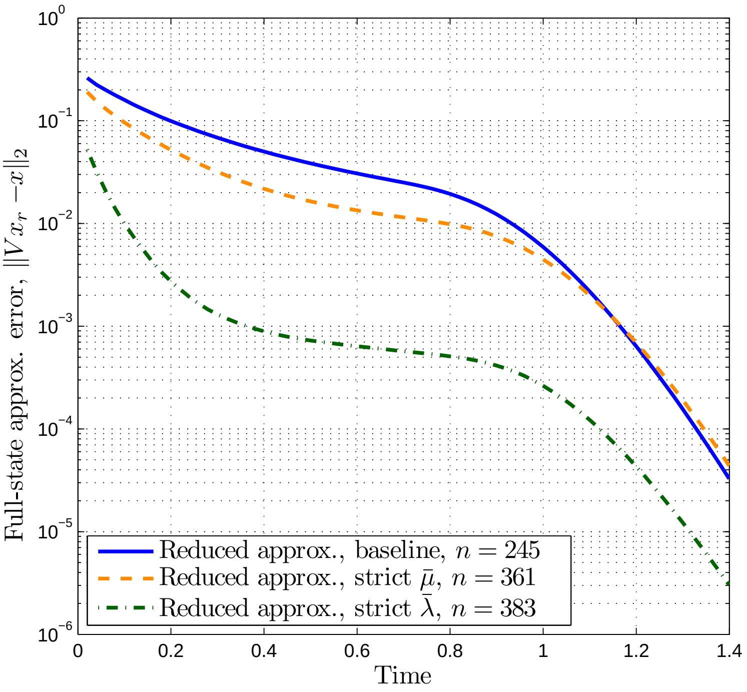


Figure 5-6: Full-state approximation error with three different reduced-order models with identical initial conditions. $Pe = 100$, $Q = 10$ outputs.

of H which are used as seed initial conditions for reduced basis construction. Fewer sensors in the contaminant transport problem will, in general, lead to a basis which is less capable of accurate full-state approximation.

Figure 5-7 illustrates the difference between x and x_r at a given time instant during the forward solve and also provides a plot of the error in the domain. The maximum error, which is approximately 15% of the maximum concentration, is localized near the edge of the domain.

5.2.2 State Approximation in the Inverse Problem Context

If the full-state approximation is to be used in conjunction with the reduced solution of the inverse problem, the reduced-order system must be solved forward in time starting from initial condition x_0^{rom}: neither the actual nor the truth initial conditions are available. Thus, in addition to the full-state approximation error discussed in the previous section, the error in the initial condition estimate is important. We demonstrate in this section that these errors do not prevent the reduced-order inversion and state approximation from providing useful information.

The experimental approach here resembles that of the previous subsection; however, instead of using an identical initial condition for both full-order and reduced-order forward solves, the truth initial condition x_0^t and the reduced-order initial condition x_0^{rom} are used as starting points for their respective models. We compare the reduced-order inversion and forward solve with the truth case as well as with the actual evolution of system state.

Results were generated with the same three reduced-order models as in the previous section. Figure 5-8 contains both of these comparisons. As seen on the right, he reduced model formed with strict $\bar{\lambda}$ again demonstrates the smallest error in full-state approximation with respect to the truth solution. Furthermore, when compared to the actual states, the same reduced model produces errors over time that are close to the errors in the truth states. This suggests that, as long as $\bar{\lambda}$ and $\bar{\mu}$ are amply strict, the accuracy losses resulting from the use of reduced-order inversion and state approximation are minimal. Figure 5-9 shows a comparison at $t = 0.2$ of the actual

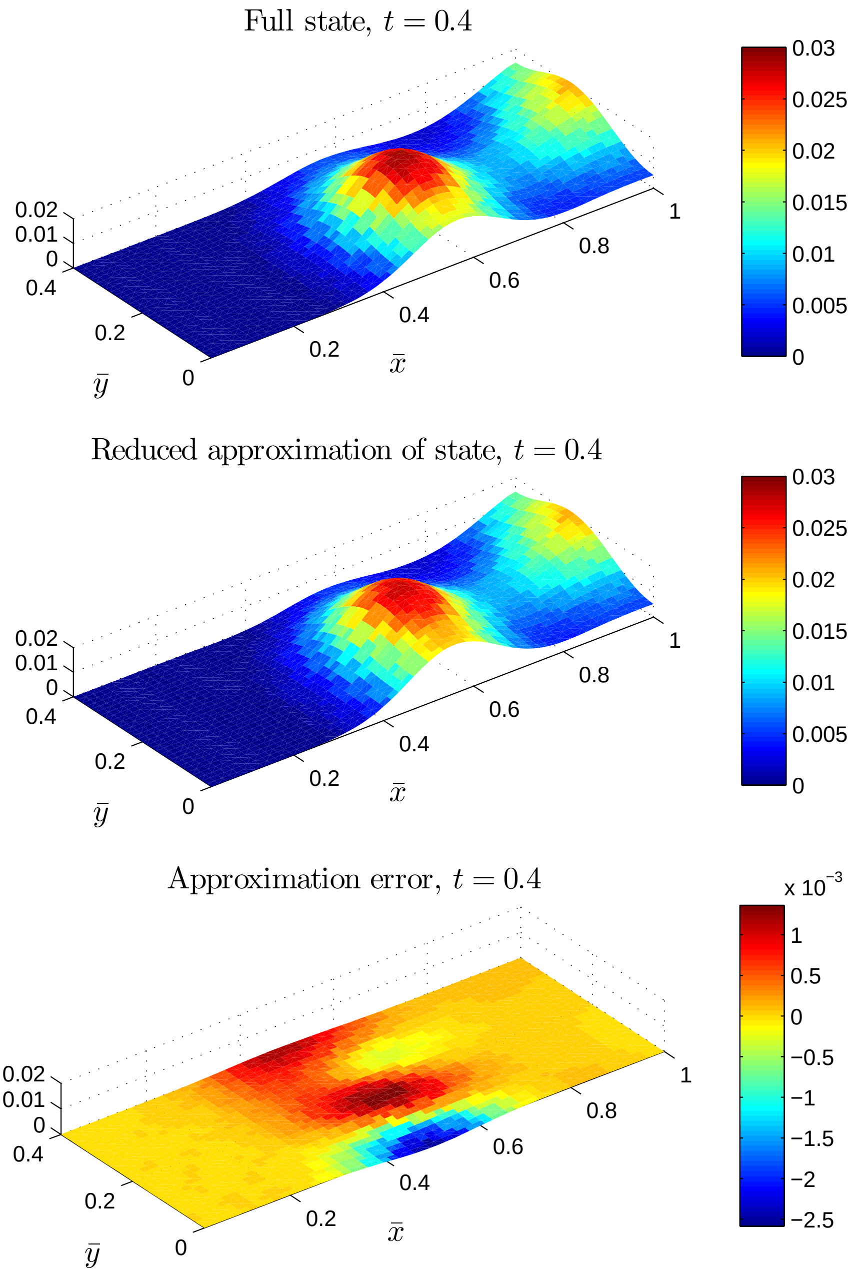


Figure 5-7: State at $t = 0.4$ as calculated by the high-fidelity (top) and by the reduced-order systems of equations (middle), along with the error between the two snapshots (bottom). The reduced model ($n = 361$) with strict $\bar{\mu}$ was used for this comparison. $Pe = 100$, $Q = 10$ outputs.

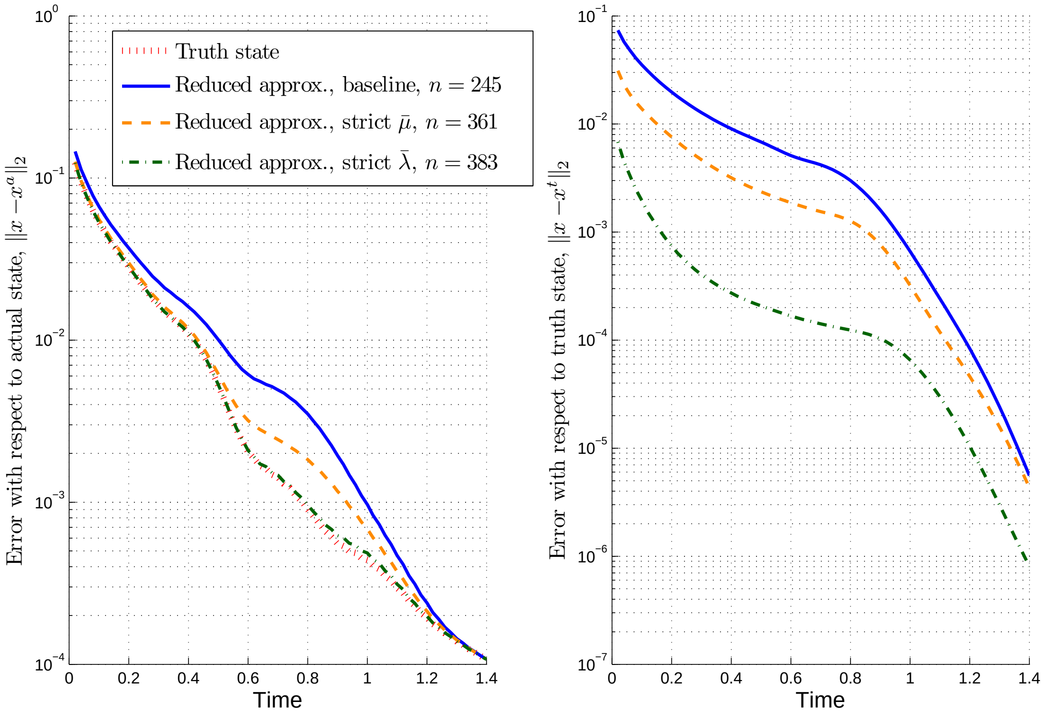


Figure 5-8: At right, full-state approximation error with respect to high-fidelity forward solve starting from the truth initial condition x_0^t. The data are shown for three different reduced-order models with their respective initial conditions x_0^{rom}. On the left, the same data plotted against the actual state evolution in the domain, including the truth case for comparison. $Pe = 100$, $Q = 10$ outputs.

state x^a, the state x^t as computed by the truth inverse solution and full-order solve, and the state x^{rom} as computed by the reduced-order processes including full-state approximation.

5.3 State Prediction from Limited Observational Data

The choice $T_{obs} = T$ in Section 5.1 was made to simplify the demonstration of inversion methods. When a *prediction* of system state is required, it is not feasible to wait T timesteps (until the system reaches steady state) to finish collecting observational data. This means that $T_{obs} < T$ in such a setting: the observation time horizon must

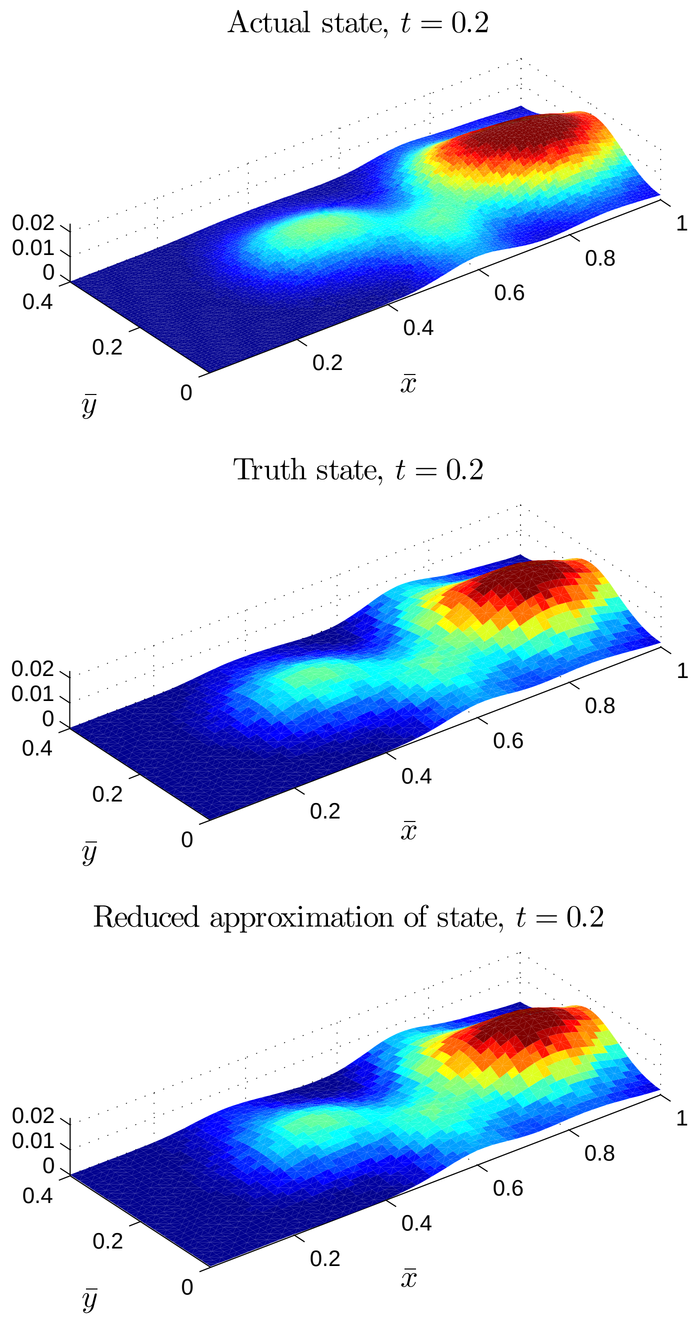


Figure 5-9: The actual state x^a in the domain at $t = 0.2$ (top); the state x^t as calculated by the high-fidelity system of equations beginning from the truth initial condition x_0^t (middle); and the state x^{rom} as calculated by the reduced-order system of equations beginning from the reduced-order initial condition x_0^{rom} (bottom). The reduced model ($n = 361$) with strict $\bar{\mu}$ was used for this comparison. Note the fine grid associated with the actual state. $Pe = 100$, $Q = 10$ outputs.

be smaller than the reduced-order model formation time horizon. In this section, we examine the relationship between T_{obs} (or t_f, the length of the time window) and the quality of the full and reduced inverse solutions. Prediction is viable if initial condition estimates based on relatively short time horizons are adequately similar to estimates based on full-length windows.

5.3.1 Time-Limited Estimates of the Initial Condition

We search for a truth initial condition $\tilde{x}_0^t$ associated with outputs $\tilde{\mathbf{y}} \in \mathbb{R}^{Q(T_{obs}+1)}$ that best match the observations $\tilde{\mathbf{y}}_{obs} \in \mathbb{R}^{Q(T_{obs}+1)}$ made during the T_{obs} timesteps in the measurement window. The modified notation from that found in (5.1) is introduced to differentiate those quantities based on a limited time horizon. Using these time-limited quantities, the inverse optimization problem is given by

$$\tilde{x}_0^t = \arg\min \; \left(\tilde{\mathbf{y}} - \tilde{\mathbf{y}}_{obs}\right)^T \left(\tilde{\mathbf{y}} - \tilde{\mathbf{y}}_{obs}\right) + \beta x_0^T x_0 \tag{5.14}$$

$$\begin{aligned} \text{where} \qquad \tilde{\mathbf{A}}\tilde{\mathbf{x}} &= \tilde{\mathbf{F}} x_0, \\ \tilde{\mathbf{y}} &= \tilde{\mathbf{C}}\tilde{\mathbf{x}}. \end{aligned}$$

The matrices $\tilde{\mathbf{A}} \in \mathbb{R}^{N(T_{obs}+1)\times N(T_{obs}+1)}$, $\tilde{\mathbf{F}} \in \mathbb{R}^{N(T_{obs}+1)\times N}$, and $\tilde{\mathbf{C}} \in \mathbb{R}^{Q(T_{obs}+1)\times N}$ have the same form as $\mathbf{A}$, $\mathbf{F}$, and $\mathbf{C}$ in (2.11) except that sizes of the former are defined in terms of T_{obs} instead of T. Similarly, $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{y}}$ take the form

$$\tilde{\mathbf{x}} = \begin{bmatrix} x(0) \\ x(1) \\ \vdots \\ x(T_{obs}) \end{bmatrix}, \qquad \tilde{\mathbf{y}} = \begin{bmatrix} y(0) \\ y(1) \\ \vdots \\ y(T_{obs}) \end{bmatrix}. \tag{5.15}$$

Following the same process as in Section 5.1, an expression to find the truth initial condition can be written in terms of the full-order, time-limited Hessian matrix

$\tilde{H} = (\tilde{\mathbf{C}}\tilde{\mathbf{A}}^{-1}\tilde{\mathbf{F}})^T(\tilde{\mathbf{C}}\tilde{\mathbf{A}}^{-1}\tilde{\mathbf{F}})$ as

$$\left(\tilde{H} + \beta I\right)\tilde{x}_0^t = (\tilde{\mathbf{C}}\tilde{\mathbf{A}}^{-1}\tilde{\mathbf{F}})^T\tilde{\mathbf{y}}_{obs}. \tag{5.16}$$

The reduced estimate of the initial condition under time-limited circumstances follows from the formulation of the truth estimate. The matrices $\tilde{\mathbf{A}}_r \in \mathbb{R}^{n(T_{obs}+1)\times n(T_{obs}+1)}$ and $\tilde{\mathbf{C}}_r \in \mathbb{R}^{Q(T_{obs}+1)\times n}$ share the structure in (2.15); the matrix $\tilde{\mathbf{f}}_r \in \mathbb{R}^{n(T_{obs}+1)\times n}$ has the structure given by (5.7). The difference is that the number of block entries in each matrix is determined by T_{obs} instead of T. Recall that all block entries in $\mathbf{A}_r$ and $\mathbf{C}_r$ depend on the reduced basis V, formed with data from $T+1$ timesteps. Thus, while the time-limited quantities contain fewer blocks because $T_{obs} < T$, data from the complete time horizon influences each block. This is possible because the reduced basis is formed offline from $T+1$ snapshots of system state.

Using $\tilde{H}_r = (\tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^{-1}\tilde{\mathbf{f}}_r)^T(\tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^{-1}\tilde{\mathbf{f}}_r)$ and a similar approach as found in Section 5.1.2, we can write two expressions that provide the reduced-order, time-limited estimate $\tilde{x}_0^{rom}$ of the initial condition:

$$\left(\tilde{H}_r + \beta I\right)\tilde{x}_{r0}^{rom} = (\tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^{-1}\tilde{\mathbf{f}}_r)^T\tilde{\mathbf{y}}_{obs}, \tag{5.17}$$

$$\tilde{x}_0^{rom} = V\tilde{x}_{r0}^{rom}. \tag{5.18}$$

5.3.2 Time-Limited Inversion Results

In this section, the effect of using limited observational data is assessed. Specifically, we use the contaminant release framework to demonstrate how the quality of the initial condition estimate varies with the length of the measurement time window. This analysis is done for both full-order and reduced-order models.

The length of the observational time window or time horizon $t_f = 1.4$ has been used throughout this work, both for snapshot generation and for demonstration of inverse problem solution. It is roughly an upper bound to the time that contaminant

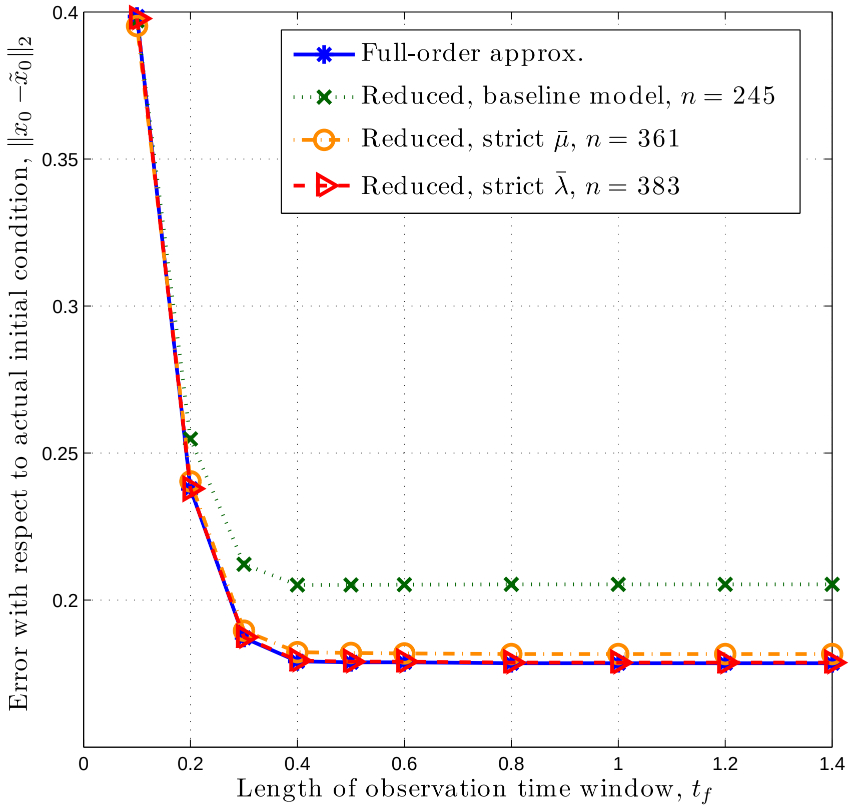
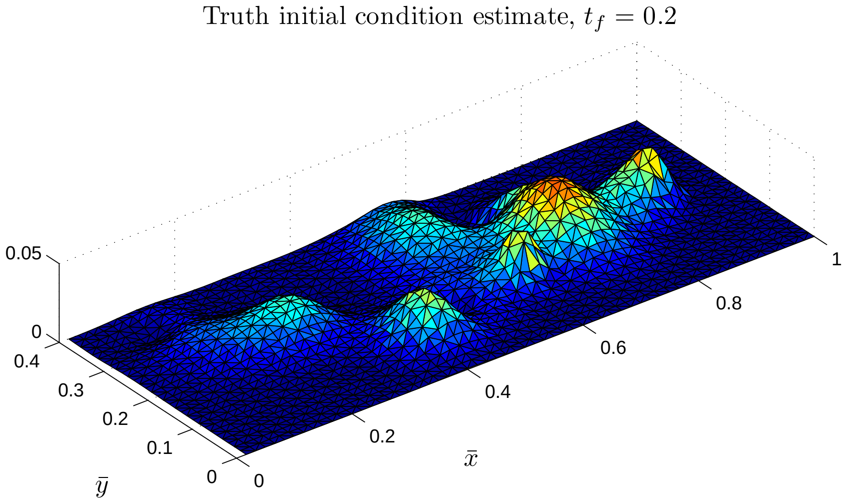


Figure 5-10: Effect of varying the length of observation time on the error between estimated initial condition and actual initial condition. The baseline and strict reduced-order models are described in Section 5.1.3. $Pe = 100$, $Q = 10$ outputs.

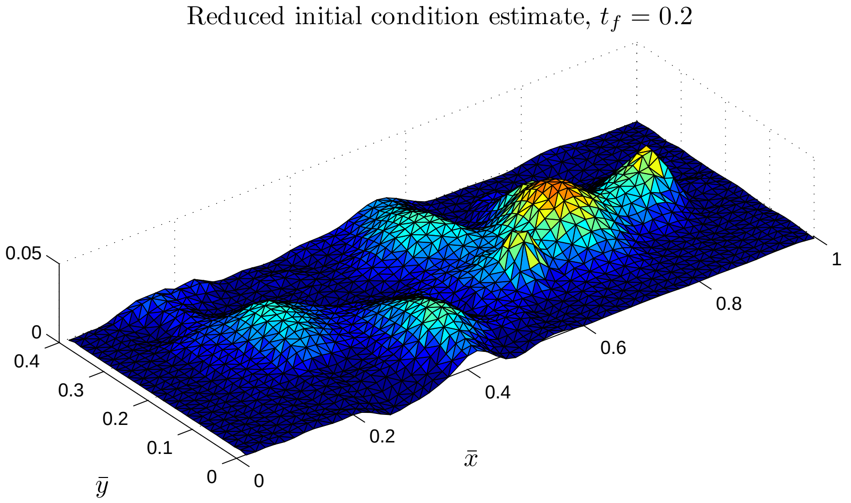
from any initial condition will spend inside the domain. If, as described in the previous section, the entire time window cannot be utilized for the collection of sensor measurements, it is necessary to rely on a lower value of t_f. With a fixed timestep $\Delta t = 0.02$, we choose different observation window lengths to form time-limited estimates $\tilde{x}_0^t$ and $\tilde{x}_0^{rom}$ of the actual initial condition x_0^a in the domain. The initial condition used for this experiment is the same as that shown in Figure 5-1, and we follow the measurement and inversion process described in Section 5.1.3 except that t_f is varied here. The results are shown in Figure 5-10.

The error for all initial condition estimates becomes smaller as the time window is lengthened and the sensors provide increasingly more data. Furthermore, the trends for both full and reduced models are similar. Because Figure 5-5 shows that all contaminant from the actual initial condition moves past the sensors by time $t = 1.0$, we expect no added accuracy for $t_f > 1.0$. In fact, we see in Figure 5-10 that increasing the window length above $t_f = 0.5$ does not appreciably change the error associated with any of the models: time windows of length $t_f = 0.5$ and $t_f = 1.4$ provide approximately identical estimates of the initial condition. Although these observations are dependent on the actual initial condition used in this experiment, they imply that time windows which are substantially shorter than the theoretical maximum are acceptable for accurate inversion. See Figure 5-11, which shows estimates of the initial condition made with $t_f = 0.2$, a factor of 7 smaller than the time horizon used to generate Figure 5-4. The results suggest that predictions of system state past the time horizon t_f can indeed be made accurately.

In addition, it may be practical to use smaller, less accurate reduced-order models if the time window desired is relatively short. Figure 5-10 demonstrates that the error associated with the baseline reduced model approaches the errors associated with higher-fidelity models as t_f is decreased. Thus, if the time window must be short in a practical setting, the accuracy benefit of using larger reduced models for inversion may not make up for the added computational cost of doing so.



Truth initial condition estimate, $t_f = 0.2$



Reduced initial condition estimate, $t_f = 0.2$

Figure 5-11: Full-order (truth) and reduced order estimates $\tilde{x}_0^t$ and $\tilde{x}_0^{rom}$ of the initial condition given a time window of length $t_f = 0.2$. Compare to Figure 5-4, which shows inversion results using a much longer time horizon of $t_f = 1.4$. The reduced-order model used is the baseline model with $\bar{\lambda} = 0.1$, $\bar{\mu} = 10^{-4}$, and $n = 245$. $Pe = 100$, $Q = 10$ outputs.

Chapter 6

Conclusion

6.1 Summary

A new method has been proposed for constructing reduced-order models of linear systems that are parametrized by initial conditions of high dimension. Formulating the greedy approach to sampling as a model-constrained optimization problem, we show that the dominant eigenvectors of the resulting Hessian matrix provide an explicit solution to the greedy optimization problems. This result leads to an algorithm to construct the reduced basis in an efficient and systematic way, and further provides quantification of the worst-case error in reduced model output prediction. Thus, the resulting reduced models are guaranteed to provide accurate replication of full-scale output quantities of interest for any possible initial condition. The worst-case error for a given reduced model can be computed using an expression that involves the dominant eigenvalue of the error Hessian matrix.

We demonstrate on a 2-D model problem that roughly a tenfold reduction in number of states still allows for accurate output replication. In the majority of experiments, the error associated with the reduced outputs is well below the computed worst-case bound. A similar 3-D experiment shows more drastic results: full-order outputs are accurately reproduced using a reduced model with 137 unknowns in place of a high-fidelity model with 65,600 unknowns. The solution of inverse problems demonstrates that the reduced models can be used to efficiently estimate initial conditions, even when measurements of system state can only be collected during a short time window. Furthermore, we show that once initial condition estimates are found, forward simulations can be performed in reduced space. The full state at each time can be approximated using the reduced state and the basis.

6.2 Future Work

The adaptive greedy sampling approach combined with the model-constrained optimization formulation provides a general framework that is applicable to nonlinear problems, although the explicit solution and maximal error guarantees apply only in the linear case. The application of the Hessian-based approach to build efficient reduced-order models for nonlinear systems has yet to be demonstrated.

We note that the task of sampling system inputs (which here were taken to be zero) to build a basis over the input space could also be formulated as a greedy optimization problem. This would allow for the construction of reduced models capable of handling inputs of high dimension.

The work presented in Chapter 4 assumes random sensor locations; however, it may be possible to optimize the positions of the sensors. For example, there may exist a sensor configuration which provides the most accurate prediction of contaminant concentration over time at certain points of interest in the domain. The sensors might also be placed in a manner that minimizes the uncertainty in the initial condition estimate. The dynamic steering of sensors based on preliminary initial condition estimates may be possible as well.

Bibliography

[1] S.S. Adavani and G. Biros. Multigrid solvers for inverse problems with parabolic PDE constraints. Submitted for publication.

[2] K. Afanasiev and M. Hinze. Adaptive control of a wake flow using proper orthogonal decomposition. In *Lecture Notes in Pure and Applied Mathematics*, volume 216, pages 317–332. Marcel Dekker, 2001.

[3] V. Akçelik, G. Biros, O. Ghattas, J. Hill, D. Keyes, and B. van Bloemen Waanders. Parallel algorithms for PDE-constrained optimization. In M. Heroux, P. Raghaven, and H. Simon, editors, *Frontiers of Parallel Computing*. SIAM, 2006.

[4] A. Antoulas. *Approximation of Large-Scale Dynamical Systems*. Advances in Design and Control DC-06. SIAM, Philadelphia, 2005.

[5] E. Arian, M. Fahl, and E.W. Sachs. Trust-region proper orthogonal decomposition for optimal flow control. Technical Report ICASE 2000-25, Institute for Computer Applications in Science and Engineering, May 2000.

[6] S. Balay, K. Buschelman, V. Eijkhout, W. Gropp, D. Kaushik, M. Knepley, L. McInnes, B. Smith, and H. Zhang. PETSc users manual. Technical Report ANL-95/11 - Revision 2.1.5, Argonne National Laboratory, 2004.

[7] S. Balay, K. Buschelman, W. Gropp, D. Kaushik, M. Knepley, L. McInnes, B. Smith, and H. Zhang. PETSc Web page, 2001. http://www.mcs.anl.gov/petsc.

[8] S. Balay, W. Gropp, L. McInnes, and B. Smith. Efficient management of parallelism in object oriented numerical software libraries. In E. Arge, A. M. Bruaset, and H. P. Langtangen, editors, *Modern Software Tools in Scientific Computing*, pages 163–202. Birkhäuser Press, 1997.

[9] G. Berkooz, P. Holmes, and J.L. Lumley. The proper orthogonal decomposition in the analysis of turbulent flows. *Annual Review of Fluid Mechanics*, 25:539–575, 1993.

[10] A.N. Brooks and T.J.R. Hughes. Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible navier-stokes equations. *Computer Methods in Applied Mechanics and Engineering*, pages 199–259, 1990.

[11] E.A. Christensen, M. Brøns, and J.N. Sørensen. Evaluation of proper orthogonal decomposition-based decomposition techniques applied to parameter-dependent nonturbulent flows. *SIAM J. Sci. Comput.*, 21(4):1419–1434, 2000.

[12] D.L. Colton and R. Kress. *Integral Equation Methods in Scattering Theory*. Pure and Applied Mathematics. John Wiley & Sons, New York, 1983.

[13] D.N. Daescu and I.M. Navon. A dual-weighted approach to order reduction in 4D-Var data assimilation. *Monthly Weather Review*, 2007. To appear.

[14] D.N. Daescu and I.M. Navon. Efficiency of a POD-based reduced second-order adjoint model in 4D-Var data assimilation. *International Journal for Numerical Methods in Fluids*, 53:985–1004, February 2007.

[15] L. Daniel, O.C. Siong, L.S. Chay, K.H. Lee, and J. White. Multiparameter moment matching model reduction approach for generating geometrically parameterized interconnect performance models. *Transactions on Computer Aided Design of Integrated Circuits*, 23(5):678–693, May 2004.

[16] A.E. Deane, I.G. Kevrekidis, G.E. Karniadakis, and S.A. Orszag. Low-dimensional models for complex geometry flows: Application to grooved channels and circular cylinders. *Phys. Fluids*, 3(10):2337–2354, 1991.

[17] B. Farrell and P. Ioannou. Accurate low-dimensional approximation of the linear dynamics of fluid flow. *Journal of the Atmospheric Sciences*, 58:2771–2789, September 2001.

[18] P. Feldmann and R.W. Freund. Efficient Linear Circuit Analysis by Padé Approximation via the Lanczos Process. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 14:639–649, 1995.

[19] K. Gallivan, E. Grimme, and P. Van Dooren. Padé Approximation of Large-Scale Dynamic Systems with Lanczos Methods. Proceedings of the 33rd IEEE Conference on Decision and Control, December 1994.

[20] G.H. Golub and C.F. Van Loan. *Matrix Computations.* Johns Hopkins Univ. Press, Baltimore, 3rd edition, 1996.

[21] M.D. Graham and I.G. Kevrekidis. Alternative approaches to the Karhunen-Loève decomposition for model reduction and data analysis. *Comput. Chem. Eng.*, 20(5):495–506, 1996.

[22] M. Grepl and A. Patera. *A posteriori* error bounds for reduced-basis approximations of parametrized parabolic partial differential equations. *ESAIM-Mathematical Modelling and Numerical Analysis (M2AN)*, 39(1):157–181, 2005.

[23] M.A. Grepl. *Reduced-basis approximation and* a posteriori *error estimation for parabolic partial differential equations.* PhD thesis, Massachusetts Institute of Technology, 2005.

[24] E. Grimme. *Krylov Projection Methods for Model Reduction.* PhD thesis, Coordinated-Science Laboratory, University of Illinois at Urbana-Champaign, 1997.

[25] S. Gugercin and A. Antoulas. A survey of model reduction by balanced truncation and some new results. *International Journal of Control*, 77:748–766, 2004.

[26] S. Gugercin, A. Antoulas, and C. Beattie. A rational Krylov iteration for optimal H2 model reduction. Proceedings of MTNS 2006, Japan, 2006.

[27] V. Hernandez, J. Roman, and V. Vidal. SLEPc: A scalable and flexible toolkit for the solution of eigenvalue problems. *ACM Transactions on Mathematical Software*, 31(3):351–362, September 2005.

[28] M. Hinze and S. Volkwein. Proper orthogonal decomposition surrogate models for nonlinear dynamical systems: Error estimates and suboptimal control. In P. Benner, V. Mehrmann, and D. Sorensen, editors, *Dimension Reduction of Large-Scale Systems*, Lecture Notes in Computational and Applied Mathematics, pages 261–306, 2005.

[29] P. Holmes, J.L. Lumley, and G. Berkooz. *Turbulence, Coherent Structures, Dynamical Systems and Symmetry.* Cambridge University Press, Cambridge, UK, 1996.

[30] K. Kunisch and S. Volkwein. Control of Burgers' equation by reduced order approach using proper orthogonal decomposition. *Journal of Optimization Theory and Applications*, 102:345–371, 1999.

[31] S. Lall, J.E. Marsden, and S. Glavaski. A subspace approach to balanced truncation for model reduction of nonlinear control systems. *International Journal on Robust and Nonlinear Control*, 12(5):519–535, 2002.

[32] J. Li and J. White. Low rank solution of Lyapunov equations. *SIAM Journal on Matrix Analysis and Applications*, 24(1):260–280, 2002.

[33] T. Penzl. Algorithms for model reduction of large dynamical systems. *Linear Algebra and its Applications*, 415(2–3):322–343, June 2006.

[34] L. Silveira and J. Phillips. Resampling plans for sample point selection in multipoint model-order reduction. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 25(12):2775–2783, December 2006.

[35] L. Sirovich. Turbulence and the dynamics of coherent structures. Part 1: Coherent structures. *Quarterly of Applied Mathematics*, 45(3):561–571, October 1987.

[36] D.C. Sorensen and A.C. Antoulas. The Sylvester equation and approximate balanced reduction. *Linear Algebra and its Applications*, 351–352:671–700, 2002.

[37] K. Veroy and A. Patera. Certified real-time solution of the parametrized steady incompressible Navier-Stokes equations: Rigorous reduced-basis *a posteriori* error bounds. *International Journal for Numerical Methods in Fluids*, 47:773–788, 2005.

[38] K. Veroy, C. Prud'homme, D. Rovas, and A. Patera. *A posteriori* error bounds for reduced-basis approximation of parametrized noncoercive and nonlinear elliptic partial differential equations. AIAA Paper 2003-3847, Proceedings of the 16th AIAA Computational Fluid Dynamics Conference, Orlando, FL, 2003.

[39] C. Vogel. *Computational Methods for Inverse Problems.* SIAM, Philadelphia, 2002.