PRACTICAL ASPECTS OF THE IMPLEMENTATION OF REDUCED-ORDER MODELS BASED ON PROPER ORTHOGONAL DECOMPOSITION

A Dissertation

by

THOMAS ANDREW BRENNER

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

May 2011

Major Subject: Aerospace Engineering

PRACTICAL ASPECTS OF THE IMPLEMENTATION OF REDUCED-ORDER

MODELS BASED ON PROPER ORTHOGONAL DECOMPOSITION

A Dissertation

by

THOMAS ANDREW BRENNER

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Approved by:

Chair of Committee,	Paul Cizmas
Committee Members,	Leland Carlson
	Othon Rediniotis
	Jean-Luc Guermond
Head of Department,	Dimitris Lagoudas

May 2011

Major Subject: Aerospace Engineering

ABSTRACT

Practical Aspects of the Implementation of Reduced-Order

Models Based on Proper Orthogonal Decomposition. (May 2011)

Thomas Andrew Brenner, B.S.E., University of Michigan

Chair of Advisory Committee: Dr. Paul G. A. Cizmas

This work presents a number of the practical aspects of developing reduced-order models (ROMs) based on proper orthogonal decomposition (POD). ROMS are derived and implemented for multiphase flow, quasi-2D nozzle flow and 2D inviscid channel flow. Results are presented verifying the ROMs against existing full-order models (FOM).

POD is a method for separating snapshots of a flow field that varies in both time and space into spatial basis functions and time coefficients. The partial differential equations that govern fluid flow can then be projected onto these basis functions, generating a system of ordinary differential equations where the unknowns are the time coefficients. This results in the reduction of the number of equations to be solved from hundreds of thousands, or more, to hundreds or less.

A ROM is implemented for three-dimensional and non-isothermal multiphase flows. The derivation of the ROM is presented. Results are compared against the FOM and show that the ROM agrees with the FOM.

While implementing the ROM for multiphase flow, moving discontinuities were found to be a major challenge when they appeared in the void fraction around gas bubbles. A point-mode POD approach is proposed and shown to have promise. A simple test case for moving discontinuities, the first order wave equation, is used to test an augmentation method for capturing the discontinuity exactly. This approach

is shown to remove the unphysical oscillations that appear around the discontinuity in traditional approaches.

A ROM for quasi-2D inviscid nozzle flow is constructed and the results are compared to a FOM. This ROM is used to test two approaches, POD-Analytical and POD-Discretized. The stability of each approach is assessed and the results are used in the implementation of a ROM for the Navier-Stokes equations.

A ROM for a Navier-Stokes solver is derived and implemented using the results of the nozzle flow case. Results are compared to the FOM for channel flow with a bump. The computational speed-up of the ROM is discussed.

Two studies are presented with practical aspects of the implementation of POD-based ROMs. The first shows the effect of the snapshot sampling on the accuracy of the POD basis functions. The second shows that for multiphase flow, the cross-coupling between field variables should not be included when computing the POD basis functions.

to my parents, for giving me the curiousity to explore and the fortitude to persist

ACKNOWLEDGMENTS

Thank you to my advisor, Dr. Paul Cizmas for his guidance and support during my time as a graduate student. His wisdom and patience were invaluable during the countless meetings that we spent discussing my research. Thank you also to my committee members, Dr. Leland Carlson, Dr. Jean-Luc Guermond and Dr. Othon Rediniotis for investing their time and knowledge in this endeavor.

Most of the work presented herein was funded by a grant from the Department of Energy. That research was conducted under the guidance of, and in collaboration with, Dr. Tom O'Brien and Dr. Ronald Breault from the National Energy Technology Laboratory. Thank you very much for your support and many helpful discussions.

I owe a great debt of gratitude to a number of my fellow graduate students. Brian Richardson introduced me to both graduate school and Texas A&M and helped launch my current research. My officemates, Forrest Carpenter and David Liliedahl, listened to all my ideas, both good and bad, and helped me figure out which was which. Raymond Fontenot contributed greatly to developing the reduced-order models for non-isothermal and three-dimensional multiphase flow. I would also like to thank the various graduate students in our research group who have helped me over the years including Joaquin Gargoloff, Shalom Johnson, Will Carter, Greg Worley, Brian Freno, Robert Brown, Nathan Jones and Neil Matula.

My education has benefited from too many excellent teachers and professors to recognize them all properly in this space. At Ovid-Elsie I was particularly impacted by Jack Nutter, Tom Hughey, Dennis Baratono, Sue Palen, Sherman Hazelton and Karl Dahlke. At the University of Michigan I found the instruction and guidance of Dr. Peter Washabaugh, Dr. Bill Kauffman and the late Dr. Gerard Faeth to be

invaluable. Thank you to all those who have given their time to impart knowledge upon me. I am in your debt.

Finally, thank you to my parents and sister for all the love and support you have given me. I could not have done it without you.

NOMENCLATURE

Roman Letters

A	Cell face area
$\mathcal{A}$	Matrix in governing system of equations
a, b	Coefficients in MFIX governing equations
$\mathcal{B}$	Right hand side vector in governing system of equations
C	Snapshot autocorrelation matrix
c	Speed of propagation
C_{pm}	Constant pressure specific heat coefficient
D	General spatial domain
$D(u)$	General linear differential operator
D_f	Diffusion term
D_p	Solids particle diameter
E	Total energy
F	Projected differential operator
F_{gs}	Interphase drag coefficient
F_{ij}	Flux for cell i through face j
$f(\alpha)$	Nonlinear differential operator

$\vec{g}$	Acceleration due to gravity
ΔH_m	Heat of reaction
h_{s0}	Initial packed bed height
J	Jacobian matrix
$J[\varphi]$	Functional for calculus of variations
K	Diffusion flux terms
k	Heat transfer coefficient
M	Number of snapshots
m	Number of POD modes retained
N	Spatial dimension of snapshots
p	Pressure
$\vec{q}_m$	Conductive heat flux
$\vec{q}$	Vector of state variables
$\overline{\overline{R}}$	Autocorrelation matrix
Res	Residual
$\overline{\overline{S}}$	Shear force
$S_{Rc\ell}$	Radiation source term
S_{ij}	Area of face j for cell i
T	Temperature

t	Time
U	Vector of state variables
(u, v, w)	Components of the velocity vector
$\vec{V}$	Velocity vector
v	Eigenvector of C
ΔV	Infinitessimal cell volume
Δx	Cell width
$\mathbf{x}, \mathbf{x}', \mathbf{y}$	General position vectors
(x, y, z)	Cartesian coordinates

Greek and Hebrew Letters

α	Time coefficient
β	Time coefficient for discontinuity mode
γ	Ratio of specific heats
γ_{Rm}	Interphase transfer coefficient
$\delta(x)$	Dirac delta measure
δ	Small variation
δ_{ij}	Kronecker delta function
ϵ	Volume fraction
ε_m	Average least-square truncation error

ϵ_g^*	Packed bed void fraction
λ	Eigenvalue
ξ	Convection weighting factor
ρ	Density
τ	Shear stress
φ	POD basis function
ψ	General non-POD basis function
ψ	General scalar
Ω	Control volume
$\aleph$	General field variable

Superscripts

$*$	Conjugate
$'$	Spatial derivative
n	Time step
o	Old values
pm	Point mode
q	State variable
ρ	Density
T	Temperature

t	Time step
V	Velocity

Subscripts

av	Average
E, W, N, S, T, B	Neighbor cell centers
e, w, n, s, t, b	Cell face indices
g	Gas
i, j, k	Mode, snapshot, or spatial index
ℓ	Phase, g (gas) or s (solids)
m	Phase, g (gas) or s (solids)
max	Maximum
nb	Neighbor cells
P	Center of cell
rel	Relative
s	Solids
ss	General snapshot
tot	Total

Acronyms and Software

CFL	Courant-Friedrichs-Lewy condition

DVODE	ODE solver for stiff systems (software)
FOM	Full-Order Model
MFIX	Multiphase Flow with Interphase Exchanges (software)
ODE	Ordinary Differential Equation
ODEx	Reduced-Order Model for multiphase flow (software)
PDE	Partial Differential Equation
POD	Proper Orthogonal Decomposition
POD-A	Proper Orthogonal Decomposition-Analytical
POD-D	Proper Orthogonal Decomposition-Discretized
PODDEC	Utility for computing POD modes (software)
ROM	Reduced-Order Model
UNS3D	Unsteady, three-dimensional Navier-Stokes solver (software)

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

FIGURE   Page

CHAPTER I

INTRODUCTION*

A. Statement of the Problem

The objective of this research was to further the understanding of the practical aspects arising in the implementation of reduced-order models (ROMs) based on proper orthogonal decomposition (POD). An effort was made to identify the ideal sampling time of the snapshots for which the proper orthogonal decomposition is performed. The best definition for the inner product used in the POD method was studied to determine if a coupled or a split approach produces less error.

Another objective of this research was to develop methods for properly capturing moving discontinuities. These moving discontinuities are present in the gas velocity fields in the form of shocks and, in multiphase flow, exist in the void fraction at the edge of bubbles. Traditional POD implementations suffer from unphysical Gibbs' oscillations near these moving discontinuities. This research augmented the POD basis functions to capture the discontinuity exactly.

An existing two-dimensional reduced-order model for multiphase flow, ODEx, was extended to model a third spatial dimension. This includes deriving and programming a reduced form of the z-direction conservation of momentum equations for the gas and solids phases. The existing model was also updated to include fluxes through the top and bottom faces of the control volume.

―――――――

The journal model is Journal of Computational Physics.

*Part of this chapter is reprinted with permission from "Practical Aspects of the Implementation of Proper Orthogonal Decomposition"by T.A. Brenner, P.G.A. Cizmas, T.J. O'Brien and R.W. Breault, 47th AIAA Aerospace Sciences Meeting, Orlando, FL, 2009. Copyright 2009 by the authors.

Using the methods developed up to this point, ROMs for the Euler and Navier-Stokes equations were derived and implemented in a finite volume solver for turbo-machinery flow. To prepare for this, and to answer key questions about the best approach to use, a ROM for flow through a nozzle, modeled using the quasi-2D Euler equations, was also derived and implemented. In particular, two approaches for deriving a ROM were compared. In the first approach, POD-Analytical (POD-A), the governing equations were projected onto the POD basis functions and the resulting system was discretized. The second approach, POD-Discretized (POD-D), used the discretized governing equations in the projection onto the POD basis functions.

B. Background

Recent advances in computer hardware have led to a wide range of new possibilities in the computational simulation of fluid flow. Despite these advances, some flows contain sufficient complexity to make numerical simulation a challenge. High-fidelity, low cost models for these flows are a necessity for both design and control, and are a major area of current and future research.

ROMs have come into wide use in the simulation of fluid flows. The dominant spatial characteristics of the flow are identified and the weighting coefficients of these modes are then computed instead of solving the governing equations at many grid points [1]. When coupled with a suitable projection method, this allows for a reduction in the degrees of freedom of a problem from tens of thousands or more to hundreds or less. The POD method identifies the dominant spatial structures in an ensemble of observations, providing the optimal modes for a ROM [2].

Early attempts to model fluid flows with ROMs focused on solving for small perturbations around a steady nonlinear flow field [3, 4, 5, 6, 7]. Subsequently, eigen-

functions of the Stokes' operator were used to generate modes for the ROM [8]. More recently, the validity of these methods has been extended through the application of POD [9, 10, 11].

Multiphase flow in fluidized beds is an example of a flow that contains sufficient complexity to make variable interaction difficult to assess. POD has been used to identify the dominant spatial features of one such flow [12]. A ROM based on POD has been implemented to reduce the computational time needed to simulate a two-dimensional isothermal multiphase flow at minimum fluidization [13]. Various acceleration methods have been developed to make the ROM for this problem more efficient [14]

Interpolation methods have been used to calculate the temporal weighting functions [15] and to enhance the robustness of the ROM for parameter changes [16]. The estimation of error in POD-based ROMs has been assessed to determine the regions of validity for POD [17]. Specific error estimates have been given for POD-based ROMs for the Navier-Stokes equations [18] and random fuzzy variables have been used to quantify error propagation through POD-based ROMs [19].

The effect of projecting partial differential equations (PDEs) with stable numerical solution methods onto truncated bases has been carefully considered [20]. The stability requirements of the Runge-Kutta integration of the ODEs resulting from the Galerkin projection of the Navier-Stokes equations onto a suitable basis has been established [21]. Further, the stability of the POD-based ROM for the linearized Euler equations, including the effect of the boundary conditions, has been analyzed [22]. Stabilization schemes have been developed to improve the accuracy of POD-based ROMs [23, 24, 25]. Spectral viscosity methods have been used to correct long-term errors due to dissipative PDEs [26]. The addition of shift-modes has been used to improve the accuracy of models in transient flow regimes [27].

Reviews of POD-based ROMs have been presented [28, 29]. Specifically, POD based ROMs have been used to model rocket nozzles for flow control [30] and for modeling airfoil cascades in the frequency domain [31]. Models have been presented for deforming grids both with [32] and without multi-POD [33] as well as for flow over heaving [34] and deforming airfoils [35].

POD-based ROMs have been implemented for modeling aero-icing [36] as well as to study flow-fields for use in particle modeling [37]. Steady supersonic flow has been predictively modeled [38]. A finite element approach for solving Burgers' equation has also been reduced using POD [39]. Recently, adaptive POD has been developed and applied to the reaction/diffusion equations to model chemical reactions [40] and the transition to turbulence around a NACA 0012 airfoil has been modeled and analyzed using a POD-based ROM [41].

POD has been used extensively to analyze the low-dimensional characteristics of experimental data [42, 43] and computational simulations [44, 45]. The effect of spatial grid [46] and time step [47] refinement has been assessed. The challenges inherent to applying POD to problems with moving discontinuities have been considered [48]. Non-POD ROMs for moving discontinuities have been developed [49]. Very recently, methods for augmenting the POD basis [50] and adding artificial viscosity to POD-based ROMs have been proposed for modeling moving discontinuities [51].

POD-based ROMs have also been used to model non-isothermal flows [52] and methods have been devised to properly couple the energy variable [53]. They have been put to practical use in modeling the temperature field in glass furnaces [54] and, when used as part of a multi-scale model, heat transfer in computer data centers [55]. Recently, a genetic algorithm has been used to replace the Galerkin projection in the POD-based ROM to improve robustness and more easily incorporate boundary conditions [56].

Various modifications to traditional POD have been proposed. These include episodic POD [57] which seeks to improve the performance of POD-based ROMs for problems with strong periodicity. Local POD plus Galerkin projection [58] updates the POD manifold periodically to reduce error while using a coarser projection to speed computation.

POD has been used outside of fluid modeling for data analysis and processing. It has been used to identify the basis functions that best model U.S. Supreme Court decisions [59]. POD has been used in image processing, including the effects of missing pixels or gappy data [60]. More recently, POD has been used to analyze fundamental characteristics of vehicle classes for design [61].

C.   Original Contributions

The ROMs presented herein for multiphase flow are a continuation of previous work [62, 63, 64]. The ROM for non-isothermal multiphase flow was developed in conjunction with Richardson and Fontenot. The author contributed to the implementation of the computational algorithm and the development of the validation case. The author derived the ROM for three-dimensional flow and collaborated with Fontenot on its implementation and the development of a validation case.

To the best of the author's knowledge, this dissertation represents the first systematic study of the differences between the POD-A and POD-D approaches for ROMs. The augmentation of the POD bases with discontinuity modes is also original work, as is the development of point-mode POD. The author also developed a full-order model (FOM) for the transport equation and implemented a FOM for the quasi-2D Euler equations.

Presented herein is a novel study of the best placement in time of the snapshots

used to compute POD basis functions for a minimum fluidization study. Also, the author conducted a study comparing coupled POD to split POD. Although the differences between the two approaches are discussed in the literature, this is, to the best of the author's knowledge, the first study comparing the error in the two approaches.

Finally, the author has derived and implemented the first stage of a ROM for the in-house implicit finite-volume solver UNS3D. Validation results are presented for a 2D channel with a bump. Plans for continuing the development of this ROM are discussed.

D. Outline of Dissertation

Chapter II presents the POD method and describes its use in reduced-order modeling. Chapter III describes the governing equations and full-order model for multiphase flow, then derives the reduced-order models for non-isothermal and three-dimensional multiphase flows and shows validation results for both ROMs. Next, Chapter IV discusses the problems inherent to POD-based ROMs for flows with moving discontinuities. Point mode POD is proposed and illustrated for multiphase flow. A simple model problem is given, two ROMs are derived using mode augmentation and validation results are presented. Chapter V describes flow through a nozzle modelling using the quasi-2D Euler equations. Two ROMs are developed for this problem and are subsequently used to evaluate the stability of contrasting methods, POD-A and POD-D. Chapter VI presents the progress made in implementing a ROM for the Navier-Stokes solver, UNS3D. Preliminary validation results are shown for a two-dimensional channel with a bump. Chapter VII discusses some practical aspects of the implementation of POD-based ROMs, including snapshot distribution and the best form of the autocorrelation matrix. Studies are conducted to evaluate various

snapshot distributions and to compare split and coupled approaches for the autocorrelation matrix. Finally, Chapter VIII presents conclusions and future work to be completed.

CHAPTER II

REDUCED-ORDER MODELING BASED ON PROPER ORTHOGONAL

DECOMPOSITION[*]

This chapter begins with a discussion of the POD method and its extension, the method of snapshots. Next, an example is given to demonstrate the process of deriving a ROM based on POD. Two approaches for deriving a ROM, POD-A and POD-D, are introduced and the advantages of each are discussed. Finally, a method for computing reference solutions for time coefficients is derived.

A.   Proper Orthogonal Decomposition

Proper orthogonal decomposition is a procedure for extracting spatial basis functions from data that are dependent on both time and space. Consider a set of discrete snapshots of a transient function $u(\mathbf{x}, \mathbf{t})$, represented by $u(\mathbf{x}, t_i)$ for $i = 1, \ldots, M$. These observations are assumed to form a linear, finite-dimensional Hilbert space L^2 on a spatial domain D. The basis functions $\{\varphi_k(\mathbf{x})\}$ are computed such that the reconstruction

$$u(\mathbf{x}, t_i) = \sum_{k=1}^{M} \alpha_k(t_i) \varphi_k(\mathbf{x}), \qquad i = 1, \ldots, M \tag{2.1}$$

is optimal in the sense that the average least-square truncation error

$$\varepsilon_m = \left\langle \left\| u(\mathbf{x}, t_i) - \sum_{k=1}^{m} \alpha_k(t_i)\, \varphi_k(\mathbf{x}) \right\|^2 \right\rangle \tag{2.2}$$

---

is a minimum for any given number $m \leq M$. Herein $|| \cdot ||$ denotes the L^2-norm given by $||f|| = (f,f)^{\frac{1}{2}}$, where $(,)$ denotes the Euclidean inner product. $\langle \cdot \rangle$ denotes an ensemble average over the number of observations, $\langle f \rangle = \sum_{i=1}^{M} f(\mathbf{x}, t_i)/M$.

This optimization problem is equivalent to the eigenvalue problem [2]

$$\int_D \langle u(\mathbf{x})u^*(\mathbf{y}) \rangle \varphi(\mathbf{y})dy = \lambda\varphi(\mathbf{x}) \tag{2.3}$$

which is a homogeneous Fredholm integral equation of the second kind [65].

Then, the optimal basis functions can be calculated as the eigenfunctions of the kernel of the integral equation (2.3). In the finite-dimensional case this reduces to finding the eigenvectors of the autocorrelation matrix defined by

$$\overline{\overline{R}}(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{M} u(\mathbf{x}, t_i)u^T(\mathbf{y}, t_i)/M. \tag{2.4}$$

Holmes et al. [2] showed that the POD basis functions are optimal using the following approach. Again consider some ensemble of scalar fields, $\{\mathbf{u}^k\}$, $\mathbf{u} = u(\mathbf{x})$. For the work presented herein, the scalar fields will be parameterized by time, *i.e.* $u^k(\mathbf{x}) = u(\mathbf{x}, t_k)$, but other parameterizations are possible and relatively common. The inner product is defined as

$$(\mathbf{f}, \mathbf{g}) = \int f(\mathbf{x})g^*(\mathbf{x})d\mathbf{x}.$$

The goal is then to compute the set of basis functions, $\{\varphi(\mathbf{x})\}$ that maximizes the projection of $\mathbf{u}$ onto φ. That is

$$\max_{\varphi \in L^2} \frac{\langle |(\mathbf{u}, \varphi)|^2 \rangle}{||\varphi||^2}.$$

This is equivalent to maximizing $\langle |(\mathbf{u}, \varphi)|^2 \rangle$ subject to the constraint $||\varphi||^2 = 1$. To

solve this problem, define the functional

$$J[\varphi] = \langle |(\mathbf{u}, \varphi)|^2 \rangle - \lambda(\|\varphi\|^2 - 1).$$

The derivative of this functional vanishes at the extrema for any variation, $\varphi + \delta\psi \in L^2, \delta \in \mathbb{R}$:

$$\frac{d}{d\delta} J[\varphi + \delta\psi]|_{\delta=0} = 0$$

or

$$\frac{d}{d\delta}[\langle (\mathbf{u}, \varphi + \delta\psi)(\varphi, \mathbf{u}) \rangle - \lambda(\varphi + \delta\psi, \varphi + \delta\psi)]|_{\delta=0} = 0.$$

If the real part is set to zero,

$$\langle (\mathbf{u}, \psi)(\varphi, \mathbf{u}) \rangle - \lambda(\varphi, \psi) = 0.$$

Expanding this inner product,

$$\left\langle \int u(\mathbf{x})\psi^*(\mathbf{x})d\mathbf{x} \int \phi(\mathbf{x}')u^*(\mathbf{x}')d\mathbf{x}' \right\rangle - \lambda \int \phi(\mathbf{x})\psi^*(\mathbf{x})d\mathbf{x} = 0.$$

Then, noting that the $\langle \cdot \rangle$ operator commutes with the integrals,

$$\int \left[\int \langle u(\mathbf{x})u^*(\mathbf{x}') \rangle \varphi(\mathbf{x}')d\mathbf{x}' - \lambda\varphi(\mathbf{x}) \right] \psi^*(\mathbf{x})d\mathbf{x} = 0.$$

But, $\psi(\mathbf{x})$ is arbitrary so

$$\int \langle u(\mathbf{x})u^*(\mathbf{x}') \rangle \varphi(\mathbf{x}')d\mathbf{x}' = \lambda\varphi(\mathbf{x}).$$

As above, this is the Fredholm integral of the second kind and the optimal basis that we are seeking is given by the eigenfunctions of

$$R(\mathbf{x}, \mathbf{x}') = \langle u(\mathbf{x})u^*(\mathbf{x}') \rangle.$$

In the finite dimensional case, this is equivalent to the finding the eigenvectors

of the tensor product matrix

$$\mathbf{R} = \langle \mathbf{u} \otimes \mathbf{u}^* \rangle.$$

B. Method of Snapshots

The method of snapshots [66] provides an efficient method for computing the POD basis functions when the spatial dimension of the snapshots, N, exceeds the total number of snapshots, M. The snapshots, $\mathbf{u}_i \equiv u(\mathbf{x}, t_i)$, and the POD basis functions, $\varphi_{\mathbf{k}}$, span the same linear space and can therefore be expressed as a linear combination where

$$\varphi_k = \sum_{i=1}^{M} v_i^k \mathbf{u}_i \qquad k = 1, \ldots, M. \tag{2.5}$$

Substituting (2.5) into the eigenvalue problem $\overline{\overline{R}}\lambda = \varphi\lambda$ gives a new eigenvalue problem

$$C\mathbf{v} = \lambda\mathbf{v}$$

where $C_{ij} = \frac{1}{M}(\mathbf{u}_i, \mathbf{u}_j)$ and $C \in \mathbb{R}^M \times \mathbb{R}^M$. Once the eigenvectors of C are computed, the POD basis functions are given by (2.5). Herein this method was used in the software package PODDEC, due to Cizmas and Palacios [10], to compute the POD basis functions.

C. Example of a Reduced-Order Model based on Proper Orthogonal Decomposition

To illustrate the process by which a ROM is constructed, consider a simple linear partial differential equation

$$\frac{\partial \mathbf{u}}{\partial t} = \mathbf{D}(\mathbf{u}) \tag{2.6}$$

where $\mathbf{D}$ is some linear differential operator and $\mathbf{u}$ is some scalar field that varies in both time and space. The POD approximation for $\mathbf{u}$ is

$$u(\mathbf{x}, t) = \bar{u}(\mathbf{x}) + \sum_{j=1}^{M} \alpha_j(t)\varphi_j(\mathbf{x}) = \sum_{j=0}^{M} \alpha_j(t)\varphi_j(\mathbf{x}) \tag{2.7}$$

where $\bar{u}(\mathbf{x})$ is the time average of $\mathbf{u}$. For convenience in notation, $\varphi_0(\mathbf{x}) \equiv \bar{u}(\mathbf{x})$ and $\alpha_0(t) \equiv 1$. Substituting (2.7) into (2.6) and projecting the resulting equation onto the k-th POD basis function, φ_k, generates a system of k ordinary differential differential equations

$$\left(\sum_{j=1}^{m} \frac{d\alpha_j(t_i)}{dt}\varphi_j(\mathbf{x}), \varphi_k(\mathbf{x}) \right) = \left(\mathbf{D}\Big(\sum_{j=0}^{m} \alpha_j(t_i)\varphi_j(\mathbf{x}) \Big), \varphi_k(\mathbf{x}) \right).$$

Using the orthogonality of the POD basis functions, this can be simplified further to find

$$\frac{d\alpha_k}{dt} = F_k(\alpha_1, \ldots, \alpha_m), \qquad k \in [1, m] \tag{2.8}$$

where F_k is some operator consisting of the projection of the spatial derivatives of POD basis functions onto POD basis functions. The orthogonality of the POD basis functions can be expressed as

$$(\varphi_i, \varphi_j) = \delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$$

Eq. (2.8) can then be integrated with some suitable ordinary differential equation (ODE) solver.

D.   Comparison of Implementation Approaches: POD-A vs. POD-D

In the preceding example the POD approximation was substituted directly into a PDE and projected onto the POD basis functions, resulting in a system of ODEs.

The evaluation of the spatial derivatives and the integration of the ODEs were not addressed. Herein this approach will be referred to as POD-A, for POD-Analytical.

A second approach exists where instead of a PDE, the governing equation is expressed as a system of discretized equations. An example would be the system of equations obtained by applying a finite difference or finite volume discretization to a PDE. Substituting the POD approximation into this system of equations results in a system of algebraic equations where the time coefficients are the unknown quantities. Projecting this new system of equations onto the POD basis functions then results in a much smaller system of algebraic equations. This approach will be demonstrated in the derivation of a ROM for multiphase flow and will be referred to as POD-D, for POD-Discretized.

E.   Reference Solution: Projection of Snapshots

A method is needed to compare the snapshots of the field variables computed using the FOM to the time coefficients computed using the ROM. Recall that a snapshot of a generic field variable u at time t_{ss} can be expressed using the POD approximation

$$u(\mathbf{x}, t_{ss}) = \varphi_0(\mathbf{x}) + \sum_{j=1}^{m} \alpha_j(t_{ss}) \varphi_j(\mathbf{x}). \tag{2.9}$$

Then, projecting both sides of (2.9) onto the POD basis functions

$$\Big(u(\mathbf{x}, t_{ss}) - \phi_0(\mathbf{x}), \varphi_k(\mathbf{x}) \Big) = \Big(\sum_{j=1}^{m} \alpha_j(t_{ss}) \varphi_j(\mathbf{x}), \varphi_k(\mathbf{x}) \Big).$$

Taking advantage of the orthogonality of the POD basis functions,

$$\alpha_k(t_{ss}) = \Big(u(\mathbf{x}, t_{ss}) - \phi_0(\mathbf{x}), \varphi_k(\mathbf{x}) \Big) \tag{2.10}$$

The projection equation (2.10) provides a method for computing the best-case time coefficients directly from the snapshots used to generate the POD basis functions. This is very useful for evaluating validation results for ROMs and will be used extensively herein.

CHAPTER III

MULTIPHASE FLOW*

This chapter begins with the hydrodynamic model used to represent multiphase flow. A computational algorithm for solving this model is then described, followed by the presentation of a reduced-order model for the non-isothermal form of this algorithm. A section is devoted to describing the model problem used the validate the non-isothermal ROM. Next a ROM for three-dimensional multiphase flow is derived. Finally, the three-dimensional model problem used to validate the 3D ROM is given.

A. Physical Model

The fluidized bed was modeled using a two-phase hydrodynamic model [67]. The governing equations were based on the laws of mass, momentum and energy conservation. In this model, the gas- and solids-phase mass balance equations are given by

$$\frac{\partial}{\partial t}(\epsilon_m \rho_m) + \nabla \cdot (\epsilon_m \rho_m \vec{v}_m) = 0 \tag{3.1}$$

where m denotes the phase, either gas or solids, ρ is the density, ϵ is the volume fraction, and $\vec{v}$ is the velocity vector.

The gas- and solids-phase momentum balance equations are given by

$$\frac{\partial}{\partial t}(\epsilon_m \rho_m \vec{v}_m) + \nabla \cdot (\epsilon_m \rho_m \vec{v}_m \vec{v}_m) = -\epsilon_m \nabla p_g + \nabla \cdot \overline{\overline{S}}_m + F_{gs}(\vec{v}_s - \vec{v}_g) + \epsilon_m \rho_m \vec{g}. \tag{3.2}$$

Here the first two terms of the right hand side represent the normal and shear surface

---

forces, respectively. The third term is the contribution of the drag force on the solids and the fourth term is the body force due to gravity.

The gas-phase energy balance equation is

$$\epsilon_g \rho_g C_{pg} \left(\frac{\partial T_g}{\partial t} + \vec{v}_g \cdot \nabla T_g \right) = -\nabla \cdot \vec{q}_g + \gamma_g (T_s - T_g) - \Delta H_g + \gamma_{Rg}(T_{Rg}^4 - T_g^4) \quad (3.3)$$

and the solids-phase energy balance equation is given by

$$\epsilon_s \rho_s C_{ps} \left(\frac{\partial T_s}{\partial t} + \vec{v}_s \cdot \nabla T_s \right) = -\nabla \cdot \vec{q}_s + \gamma_s (T_s - T_g) - \Delta H_s + \gamma_{Rs}(T_{Rs}^4 - T_s^4), \quad (3.4)$$

where $\vec{q}_m$ is the conductive heat flux, ΔH_m is the heat of reaction and γ_{Rm} is the heat transfer coefficient. The constant pressure specific heat coefficients for the gas- and solids-phases are C_{pg} and C_{ps}, respectively.

B. Full-Order Model: MFIX

The full-order model refers to the numerical model used to solve these governing equations and generate the database of snapshots used by the POD method. The FOM was developed at the Department of Energy's National Energy Technology Laboratory and the implementation is the Multiphase Flow with Interface eXchanges (MFIX) code [67]. For isothermal cases, this code solves a discretized version of (3.2) and uses correction algorithms that satisfy (3.1) to calculate the gas pressure and solids volume fraction. For non-isothermal cases, discretized versions of (3.3) and (3.4) are also solved.

The solutions of these equations were collected throughout the time domain to form a database of snapshots for both the isothermal and non-isothermal cases. For the isothermal case, snapshots of six field variables were captured: x- and y-direction gas and solids velocities, gas pressure, and void fraction ($\epsilon = \epsilon_g = 1 - \epsilon_s$). For the

non-isothermal case, the gas and solids temperature fields were also collected.

C. Reduced-Order Model for Non-Isothermal Flow

Previous work has presented the derivation of a ROM for two-dimensional isothermal flow in fluidized beds [13, 14]. This section presents the derivation of a ROM for the gas and solids energy balance equations to be used in conjunction with the previously developed models for x- and y-direction gas and solids velocities, void fraction, and gas pressure.

The energy balance equations (3.3) and (3.4) can be written in the general form of transport equations. That is, the discretized transport equation for a scalar ψ can be written as [68, p. 18, Eq. 3.12]

$$a_P(\psi)_P = \sum_{nb} a_{nb}(\psi)_{nb} + b_P, \tag{3.5}$$

where p and nb denote the cell center and neighbor cell centers, respectively. For the energy balance equation $\psi = T_\ell$, where ℓ denotes the phase, either gas or solids. The coefficients of the discretized energy balance equation are

$$a_E^{T_\ell} = D_e - \frac{(\xi_\ell^T)_e}{2}(\epsilon_\ell\rho_\ell)_E(C_{p\ell} + C_{p\ell E})(u_\ell)_e A_e \tag{3.6}$$

$$a_W^{T_\ell} = D_w + \frac{(\bar{\xi}_\ell^T)_w}{2}(\epsilon_\ell\rho_\ell)_W(C_{p\ell} + C_{p\ell E})(u_\ell)_w A_w \tag{3.7}$$

$$a_N^{T_\ell} = D_n - \frac{(\xi_\ell^T)_n}{2}(\epsilon_\ell\rho_\ell)_N(C_{p\ell} + C_{p\ell N})(v_\ell)_n A_n \tag{3.8}$$

$$a_S^{T_\ell} = D_s + \frac{(\bar{\xi}_\ell^T)_s}{2}(\epsilon_\ell\rho_\ell)_S(C_{p\ell} + C_{p\ell N})(v_\ell)_s A_s \tag{3.9}$$

$$a_p^{T_\ell} = -\left(\sum_{nb}(a_\ell^T)_{nb} + \frac{(\rho_\ell^o)C_{p\ell}\triangle V}{\Delta t} + \gamma_{Rm}(T_\ell^o)^3\right) \tag{3.10}$$

$$b_p^{T_\ell} = -\left(\frac{(\rho_\ell^o)C_{p\ell}\triangle V}{\triangle t}(T_\ell^o) - \triangle H_{Rm}\triangle V + S_{RC\ell}\triangle V\right) \tag{3.11}$$

$$(S_{RC\ell})_p = \gamma_{Rm}(T_{Rm}^4 + 3(T_\ell^o)^4) - 4\gamma_{Rm}(T_\ell^o)^3 T_\ell \tag{3.12}$$

In (3.6)-(3.12), ℓ denotes the phase index (g or s). The subscripts E, W, N and S denote the cell centers of the East, West, North and South neighbor cells, respectively. The subscripts e, w, n and s denote the East, West, North and South face of the cell, respectively. The diffusion terms, D_f, where f is the face of the cell, are modeled as [68, p. 19, Eq. 3.23]

$$D_f = \frac{K_{\ell f} A_f}{\triangle x_f}.$$

The $K_{\ell f}$ flux terms are given in the MFIX Numerical Technique [68, p. 17, Eq. 3.6 & 3.7]. ξ, ϵ, ρ, and C_p are the convection weighting factor, gas void fraction, density and constant pressure specific heat, respectively. u_ℓ and v_ℓ are the gas and solids velocities in the x- and y-direction. A is the area of a cell face and the infinitesimal cell volume is represented by $\triangle V$. The superscript o denotes "old" variables, that is, variables from the previous time step. Herein the radiation source term, $S_{RC\ell}$, is assumed to be zero.

The temperature T_ℓ is approximated using the POD basis functions and time coefficients as

$$T_\ell(\mathbf{x}, t_k) \cong \varphi_0^{T_\ell}(\mathbf{x}) + \sum_{j=1}^{m^{T_\ell}} \alpha_j^{T_\ell}(t_k)\varphi_j^{T_\ell}(\mathbf{x}). \tag{3.13}$$

Substituting (3.13) into (3.5) yields

$$a_p\left(\varphi_0^{T_\ell}(\mathbf{x}) + \sum_{j=1}^{m^{T_\ell}} \alpha_j^{T_\ell}(t_k)\varphi_j^{T_\ell}(\mathbf{x})\right) = \sum_{nb} a_{nb}\left(\varphi_{0,nb}^{T_\ell}(\mathbf{x}) + \sum_{j=1}^{m^{T_\ell}} \alpha_j^{T_\ell}(t_k)\varphi_{j,nb}^{T_\ell}(\mathbf{x})\right) + b_P. \tag{3.14}$$

Projecting (3.14) onto the POD basis functions using a Galerkin projection gives the system of equations

$$\tilde{\mathcal{A}}^{T_\ell} \alpha^{T_\ell} = \tilde{\mathcal{B}}^{T_\ell} \tag{3.15}$$

where

$$\tilde{\mathcal{A}}_{ij}^{T_\ell} = \left(a_P^{T_\ell} \varphi_j^{T_\ell}(\mathbf{x}) - \sum_{nb} a_{nb}^{T_\ell} \varphi_{j,nb}^{T_\ell}(\mathbf{x}), \varphi_i^{T_\ell}(\mathbf{x}) \right) \tag{3.16}$$

and

$$\tilde{\mathcal{B}}_i^{T_\ell} = \left(b_P^{T_\ell} + \sum_{nb} a_{nb}^{T_\ell} \varphi_{0,nb}^{T_\ell}(\mathbf{x}) - a_P^{T_\ell} \varphi_0^{T_\ell}(\mathbf{x}), \varphi_i^{T_\ell}(\mathbf{x}) \right) \tag{3.17}$$

The dimensions of $\tilde{\mathcal{A}}^{T_\ell}$ and $\tilde{\mathcal{B}}^{T_\ell}$ are $m^{T_\ell} \times m^{T_\ell}$ and $m^{T_\ell} \times 1$, respectively.

D. Model Problem

A two-dimensional non-isothermal fluidized bed at minimum fluidization [69] was used to verify the reduced-order model for non-isothermal flow. The geometry is shown in Fig. 1. Background gas is injected along the bottom of the bed at $v_2 = 1$ cm/s. After $t = 0.2$ s, a central jet is introduced at $v_1 = 12.6$ cm/s and the combined flow is simulated for the remainder of the first second. The first 0.2 s were considered start-up transience and, as a result, only the final 0.8 s were considered for the model verification.

The bed is initially at 297 K and the distributed inlet flow is also at 297 K. The central jet, however, is at 450 K. The parameters for this test case are given in Table I.
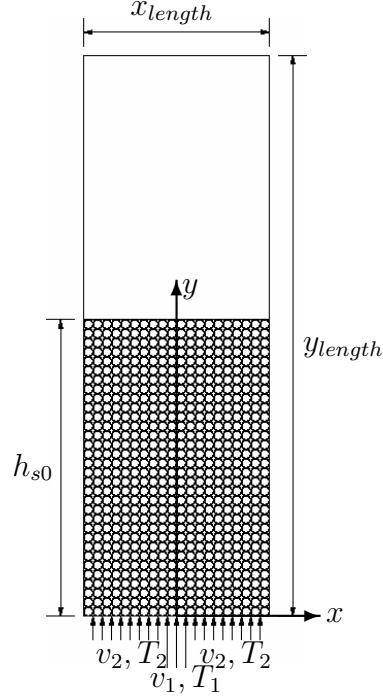
Figure 1: Geometry and boundary conditions for the two-dimensional fluidized bed.

E.   Non-Isothermal Flow Results

The non-isothermal test case was simulated using the FOM and 320 snapshots of each field variable were collected. The POD basis functions were computed using the method of snapshots and a split approach. The ROM was then used to simulate the problem at the same reference conditions using the mode-set given in Table II. The relatively large number of modes needed to compute the gas temperature was due to the gas temperature still being in a transient regime in the temporal domain considered.

Figure 2 shows the cumulative energy captured by the basis functions of the eight field variables in the non-isothermal model problem. The cumulative energy is the

summation of the relative energy captured by each POD mode. The relative energy of the k-th POD mode is defined as $\lambda_k / \sum_{i=1}^{M} \lambda_i$ where λ_k is the eigenvalue associated with the k-th POD mode in (2.3).
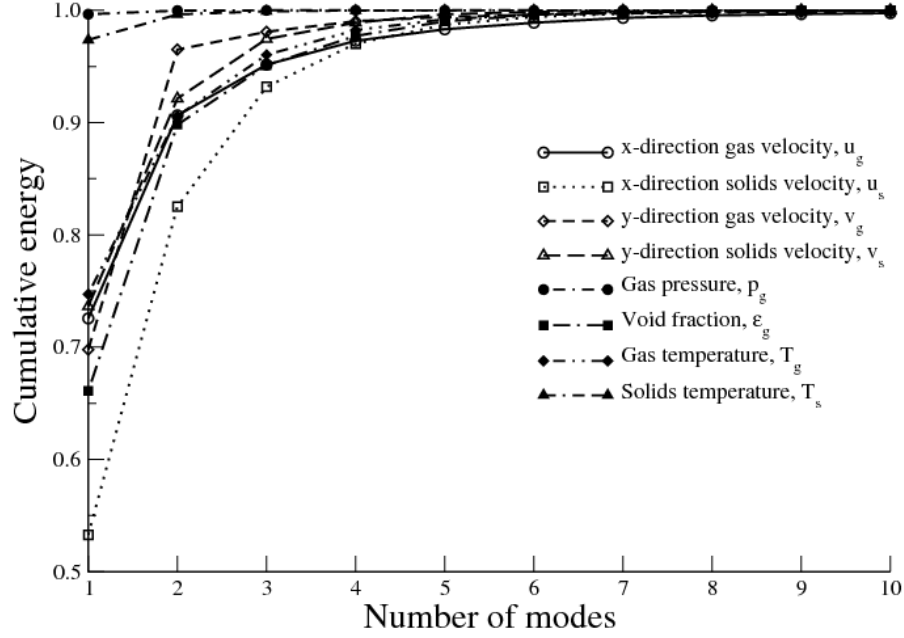
Figure 2: Cumulative energy spectrum for non-isothermal model problem.

Figures 3 and 4 compare the gas and solids temperature fields computed using the FOM and ROM at $t = 1$ s and Figs. 5 and 6 show the same fields in greater detail around the jet inlet. The agreement is very good. This agreement is also evident in Figs. 7 and 8 which compare the time coefficients for the first two POD modes for gas and solids temperature computed with the ROM to the time coefficients computed by projecting the snapshots directly onto the POD basis functions using (2.10).
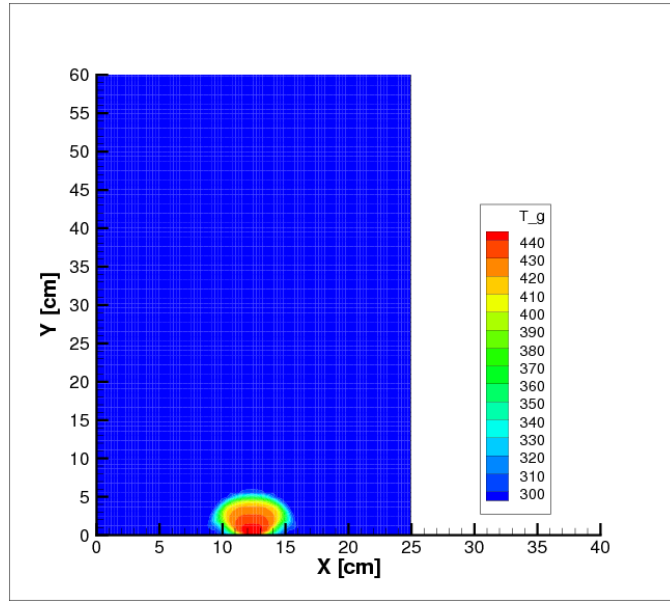
(a)
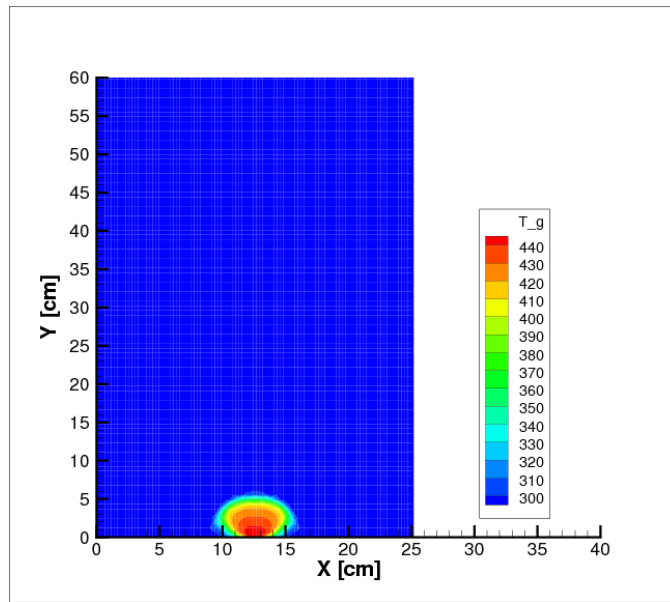
(b)

Figure 3: Contour plots at $t = 1.0$ s for: (a) gas temperature, FOM, (b) gas temperature, ROM.
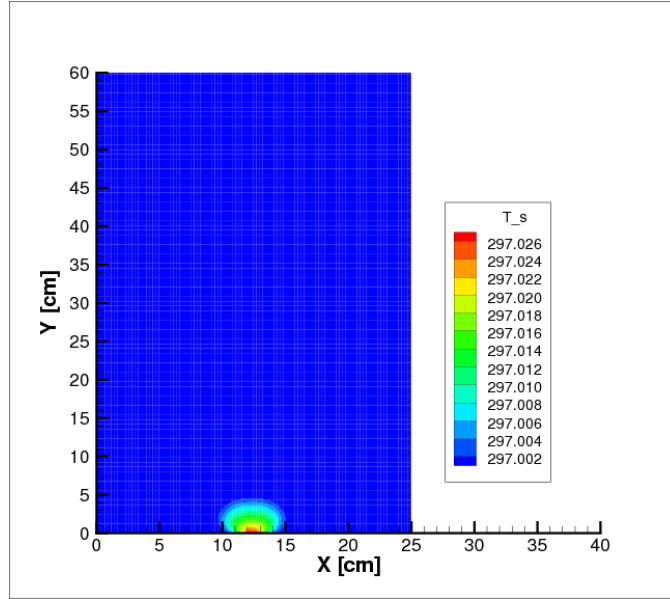
(a)
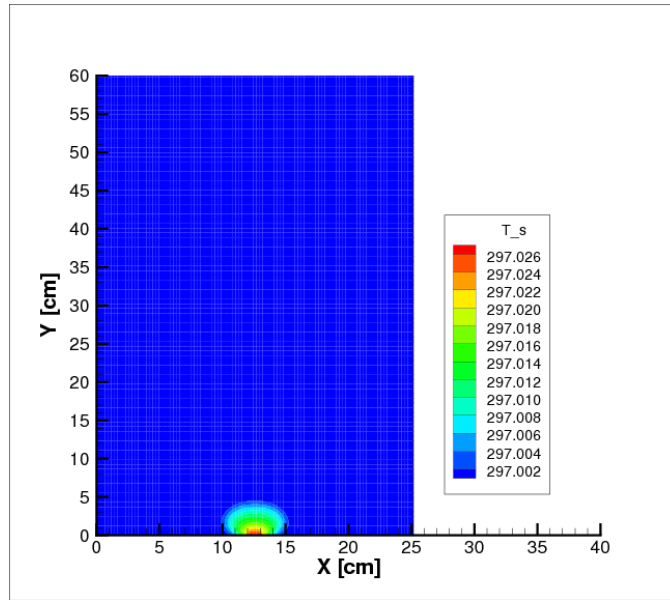
(b)

Figure 4: Contour plots at $t = 1.0$ s for: (a) solids temperature, FOM, (b) solids temperature, ROM.
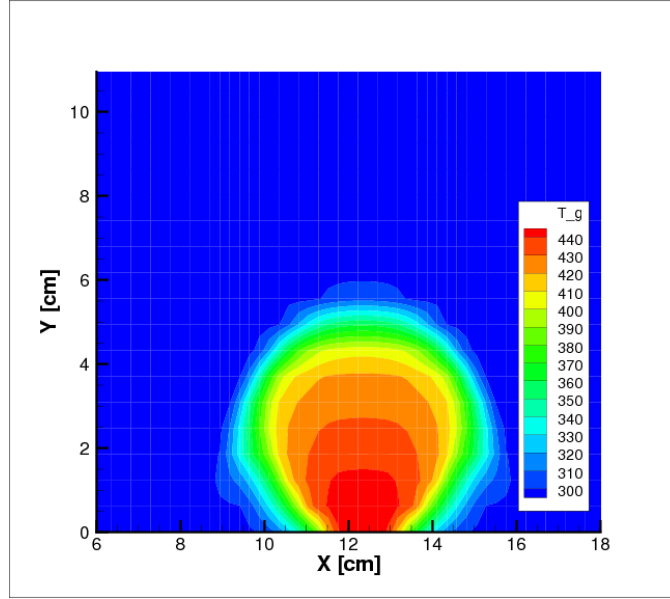
(a)
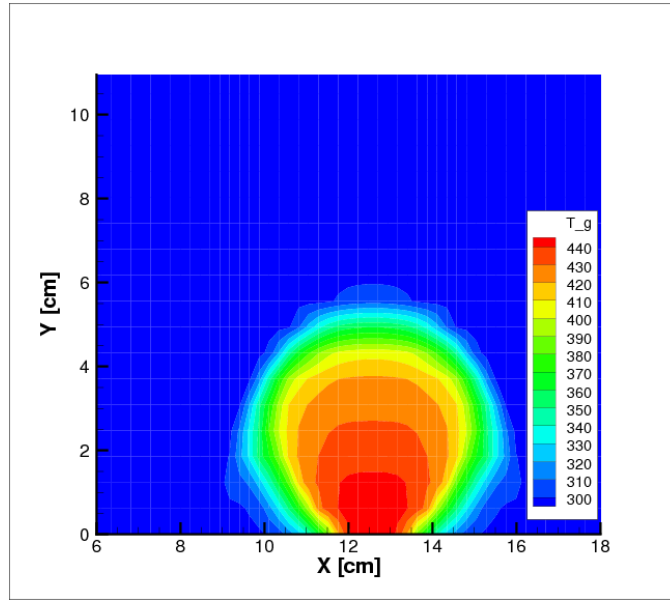
(b)

Figure 5: Contour plots showing the region around the jet inlet at $t = 1.0$ s for: (a) gas temperature, FOM, (b) gas temperature, ROM.
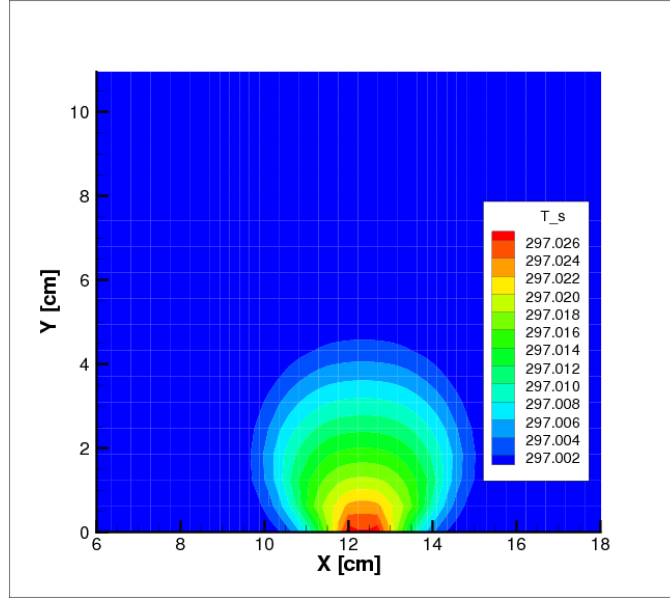
(a)
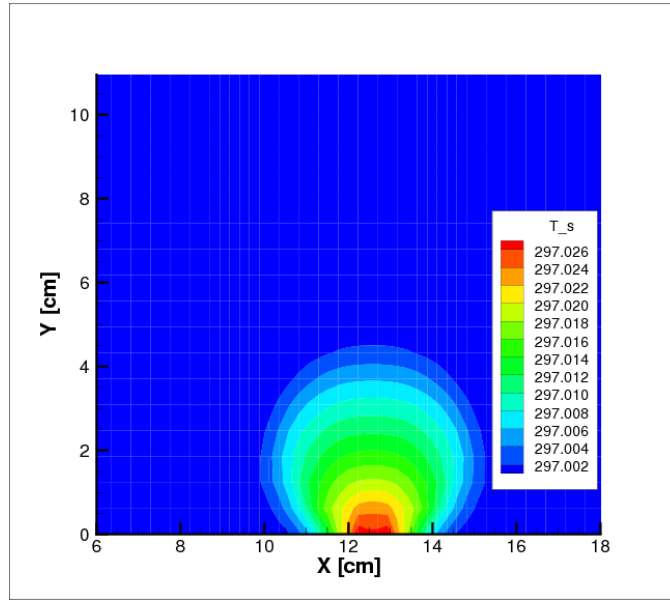
(b)

Figure 6: Contour plots showing the region around the jet inlet at $t = 1.0$ s for: (a) solids temperature, FOM, (b) solids temperature, ROM.
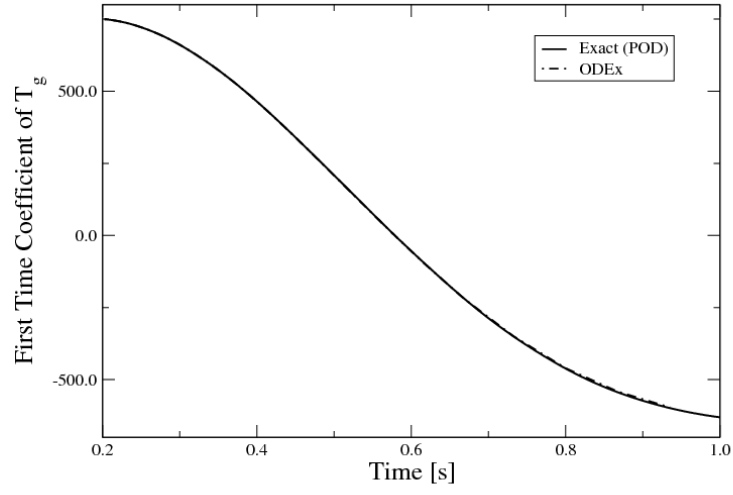
(a)
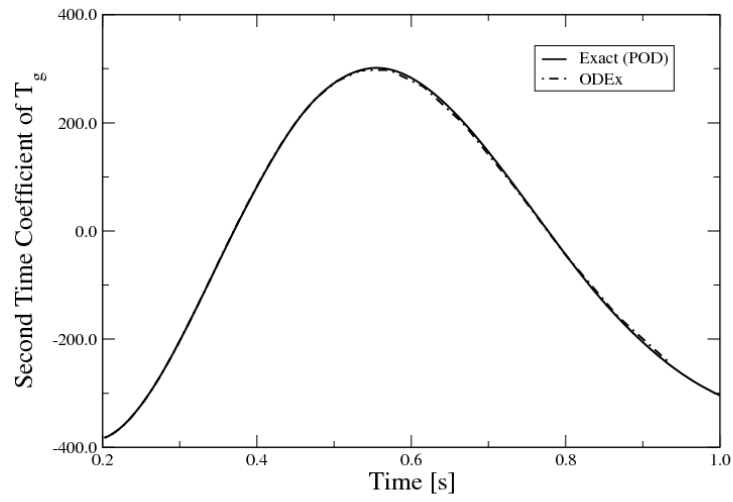
(b)

Figure 7: Comparison of time coefficients for gas temperature: (a) $\alpha_1^{T_g}$, and (b) $\alpha_2^{T_g}$.
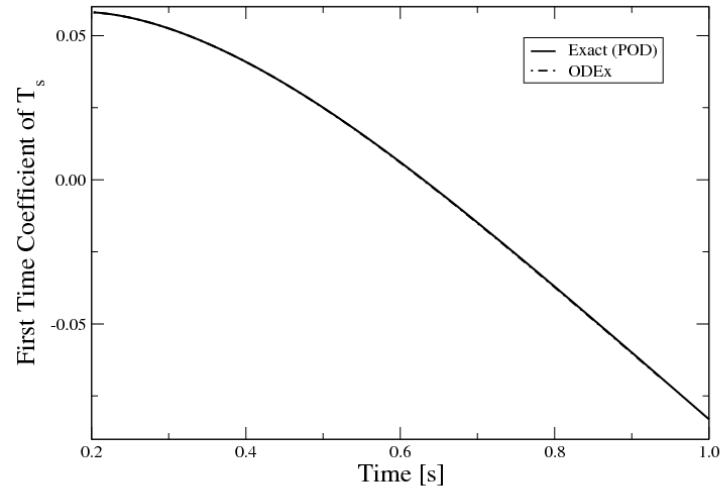
(a)
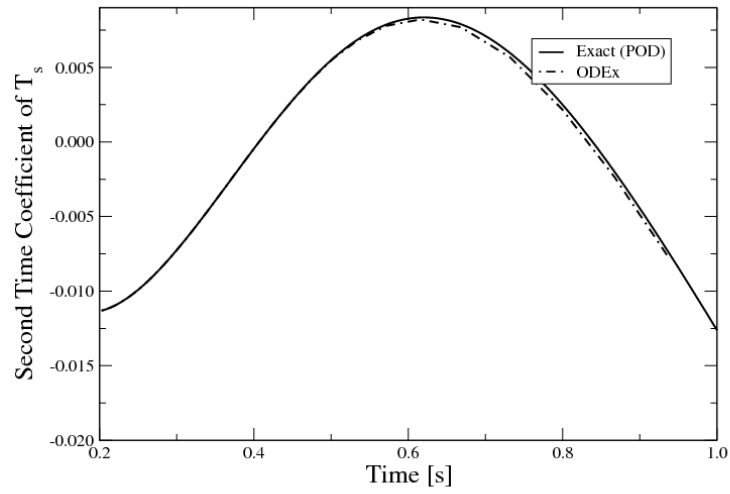
(b)

Figure 8: Comparison of time coefficients for solids temperature: (a) $\alpha_1^{T_s}$, and (b) $\alpha_2^{T_s}$.

Table I: Parameters of a non-isothermal fluidized bed [69].

Parameter	Description	Value
x_{length}	Length of the domain in x-direction	25.4 cm
y_{length}	Length of the domain in y-direction	76.5 cm
i_{max}	Number of cells in x-direction	108
j_{max}	Number of cells in y-direction	124
v_1	Jet gas inflow velocity	12.6 cm/s
v_2	Distributed gas inflow velocity	1.0 cm/s
p_s	Static gas pressure at outlet	1.01×10^6 g/(cm·s^2)
T_{g0}	Initial gas temperature	297 K
T_{s0}	Initial solids temperature	297 K
T_{g1}	Jet gas inlet temperature	450 K
T_{g2}	Distributed gas inlet temperature	297 K
μ_{g0}	Gas viscosity	1.8×10^{-4} g/(cm·s)
t_{start}	Start time	0 s
t_{stop}	Stop time	1 s
$\triangle t$	Initial time step	1.0×10^{-4} s
ρ_{so}	Constant solids density	1.0 g/cm^3
D_p	Solids particle diameter	0.5 mm
h_{s0}	Initial packed bed height	38.25 cm
C_{pg0}	Initial gas phase specific heat	0.25 cal/gK
C_{ps0}	Initial solids phase specific heat	0.310713 cal/gK
ϵ_g^*	Packed bed void fraction	0.40

Table II: Number of modes used in the non-isothermal case.

Field variable	Symbol	No. of modes
Gas pressure	m^{p_g}	2
Void fraction	m^{ϵ_g}	7
u gas velocity	m^{u_g}	2
v gas velocity	m^{v_g}	5
u solids velocity	m^{u_s}	8
v solids velocity	m^{v_s}	6
Gas temperature	m^{T_g}	12
Solids temperature	m^{T_s}	3

F. Reduced-Order Model for Three-Dimensional Flow

MFIX models the $z-$direction gas and solids velocities as

$$a_p^{w_\ell} w_\ell + \sum_{nb} a_{nb}^{w_\ell} w_\ell = b_p^{w_\ell} \tag{3.18}$$

where the $a_{nb}^{w_\ell}$, $a_p^{w_\ell}$, $b_p^{w_\ell}$ coefficients are defined in the MFIX Numerical Technique document [68, pp. 15-21] and ℓ represents the phase, either gas or solids. The velocity can then be approximated using the POD basis functions as

$$w_\ell(x,t) = \varphi_0^{w_\ell}(x) + \sum_{j=1}^{m^{w_\ell}} \alpha_j^{w_\ell}(t)\varphi_j^{w_\ell}(x), \tag{3.19}$$

where m^{w_ℓ} is the number of z-direction velocity modes kept in the approximation. Substituting (3.19) into (3.18), projecting the result onto the POD basis functions, and collecting the time coefficient terms on the left hand side of the equation,

$$\tilde{\mathcal{A}}^{w_\ell} \alpha^{w_\ell} = \tilde{\mathcal{B}}^{w_\ell},$$

where

$$\tilde{\mathcal{A}}_{ij}^{w_\ell} = \left(\left(a_p^{w_\ell} \phi_j^{w_\ell} + \sum_{nb} a_{nb}^{w_\ell} \phi_{j,nb}^{w_\ell} \right), \phi_i^{w_\ell} \right),$$

$$\tilde{\mathcal{B}}_i^{w_\ell} = (b_p^{w_\ell} - a_p^{w_\ell} - \sum_{nb} a_{nb}^{w_\ell} \phi_{0,nb}^{w_\ell}, \phi_i^{w_\ell}).$$

The dimensions of $\tilde{\mathcal{A}}$ and $\tilde{\mathcal{B}}$ are $m^{w_\ell} \times m^{w_\ell}$ and $m^{w_\ell} \times 1$, respectively. Eq. (3.20) was implemented in ODEx for both the gas and solids phases. This equation was solved immediately after the x- and y-direction conservation of momentum equations for the gas and solids phases, respectively. The solution methodology was the same as for the x- and y-direction conservation of momentum equations as described in Yuan [62] and Richardson [63]. The solution algorithms for the other field variables were updated to include flux terms through the top and bottom of the control volume.

G. 3D Results

A three-dimensional test case was developed to test the ROM. This test case used the parameters specified in Table III and the geometry shown in Fig. 1 was extruded in the z-direction. For this case the central jet was a slot oriented in the z-direction. MFIX was used to model the flow and PODDEC was used to extract the basis functions. The time coefficients computed using ODEx with the number of modes given in Table IV are shown in Figs. 9-16 where they are compared to the best-case time coefficients computed using the projection method given by (2.10). The agreement is generally very good.
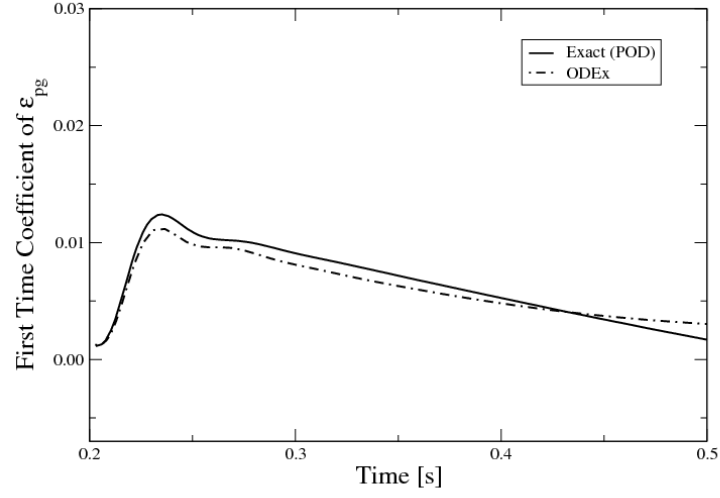
(a)
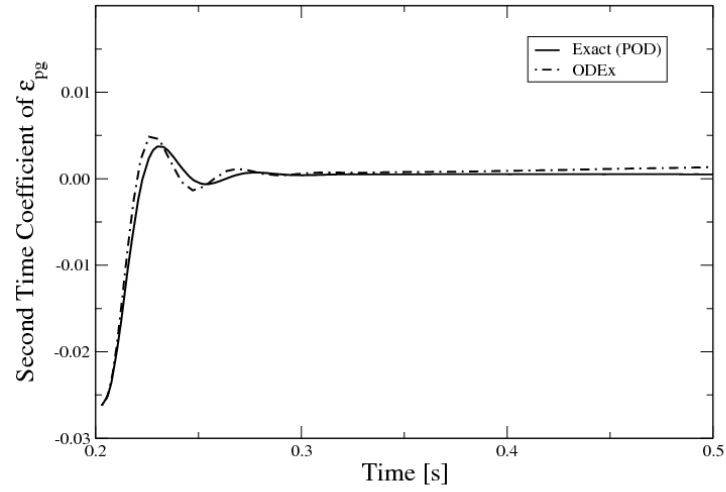
(b)

Figure 9: Time history of the first two time coefficients for void fraction in the 3D case: (a) $\alpha_1^{\epsilon_g}$, and (b) $\alpha_2^{\epsilon_g}$.
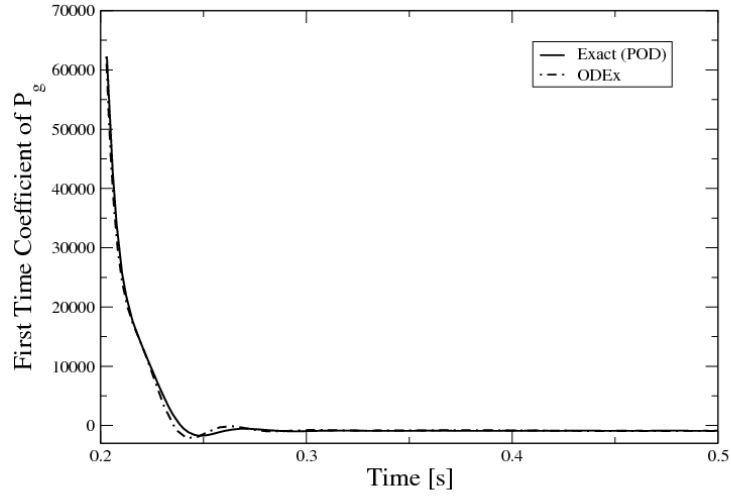
(a)
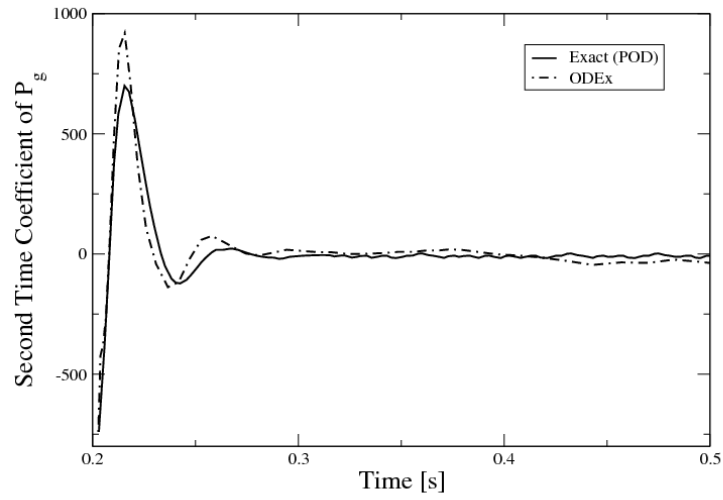
(b)

Figure 10: Time history of the first two time coefficients for gas pressure in the 3D case: (a) $\alpha_1^{p_g}$, and (b) $\alpha_2^{p_g}$.
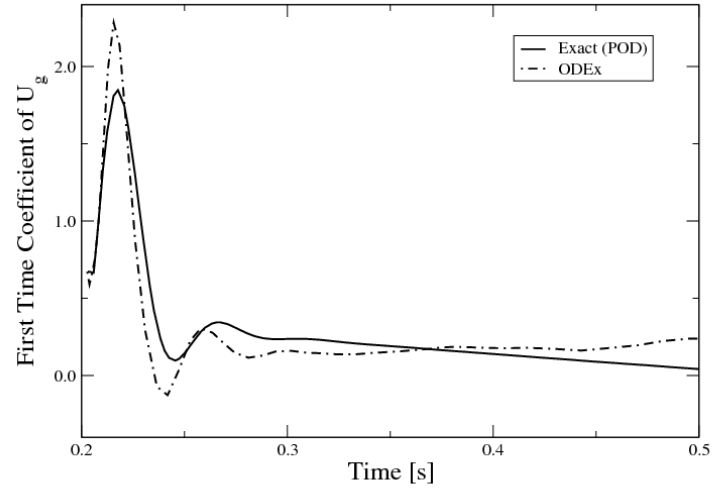
(a)
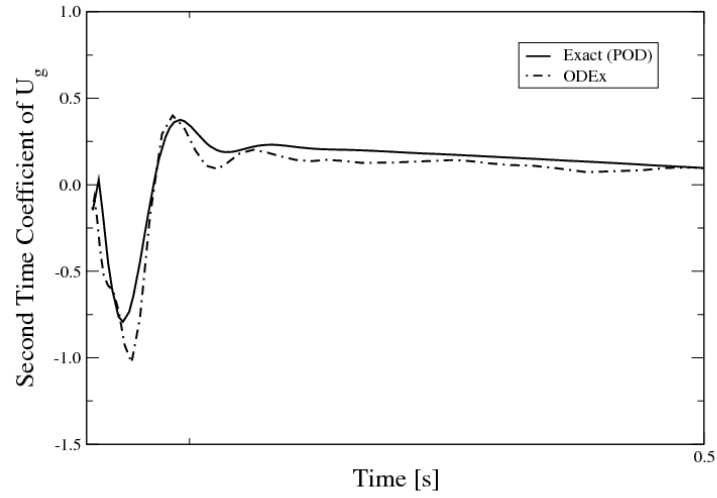
(b)

Figure 11: Time history of the first two time coefficients for x-direction gas velocity in the 3D case: (a) $\alpha_1^{u_g}$, and (b) $\alpha_2^{u_g}$.
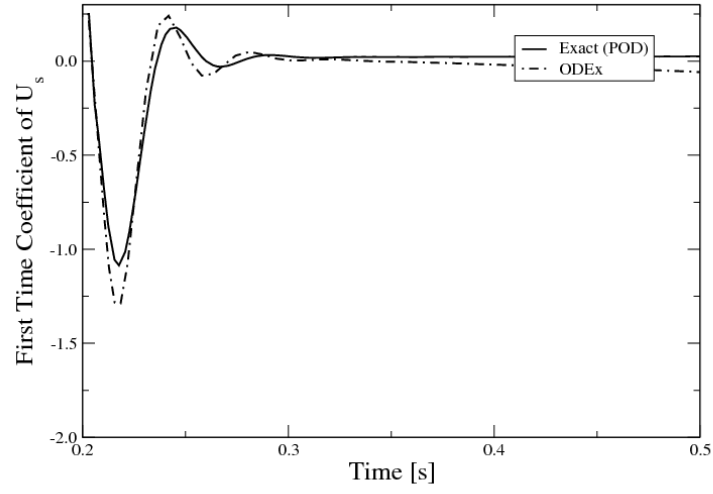
(a)
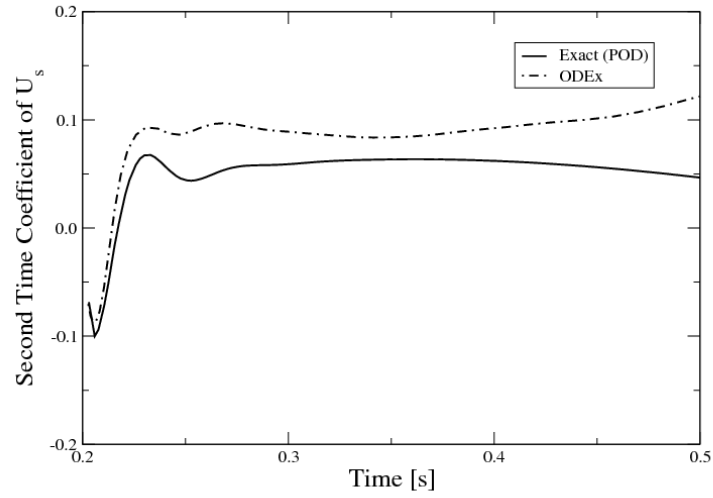
(b)

Figure 12: Time history of the first two time coefficients for x-direction solids velocity in the 3D case: (a) $\alpha_1^{u_s}$, and (b) $\alpha_2^{u_s}$.
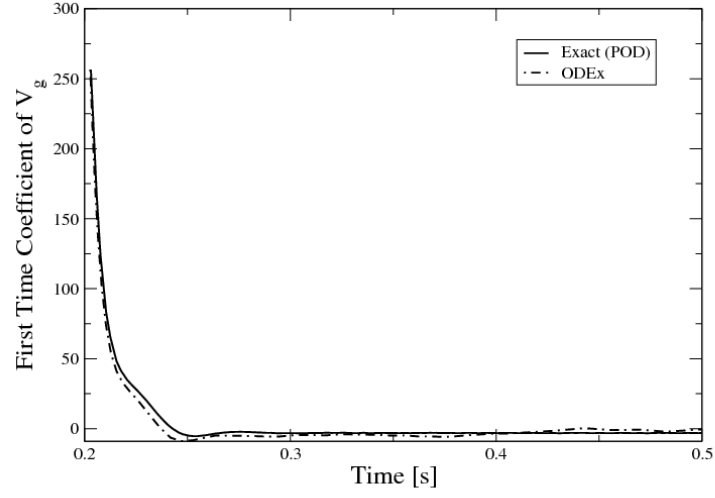
(a)
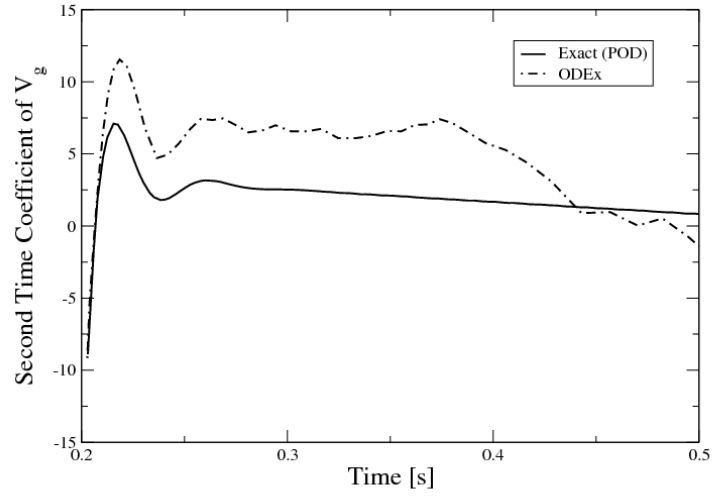
(b)

Figure 13: Time history of the first two time coefficients for y-direction gas velocity in the 3D case: (a) $\alpha_1^{v_g}$, and (b) $\alpha_2^{v_g}$.
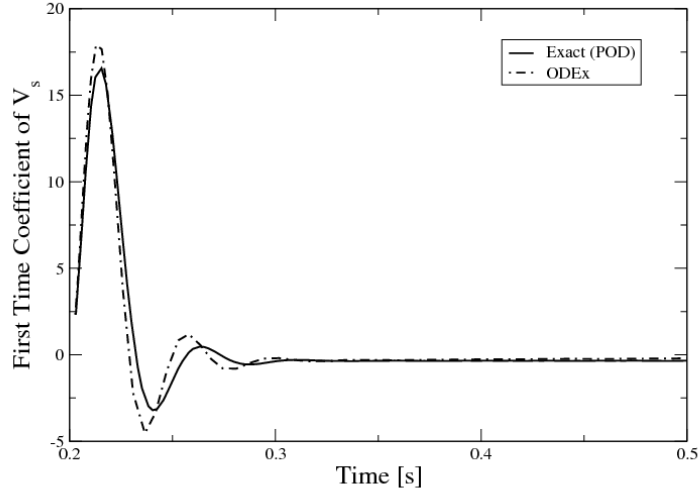
(a)
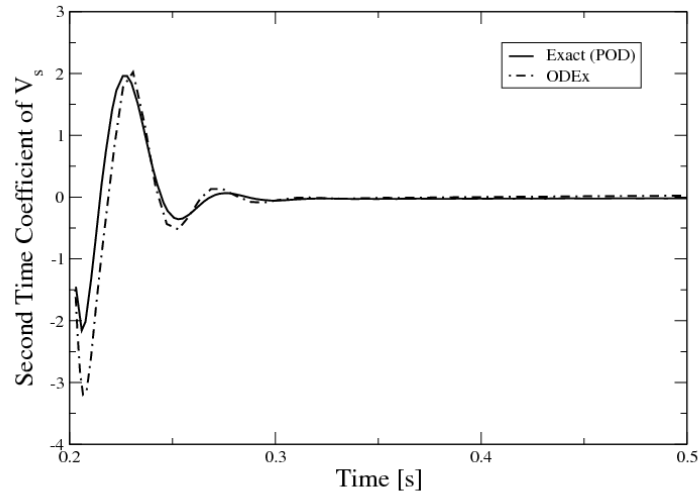
(b)

Figure 14: Time history of the first two time coefficients for y-direction solids velocity in the 3D case: (a) $\alpha_1^{v_s}$, and (b) $\alpha_2^{v_s}$.
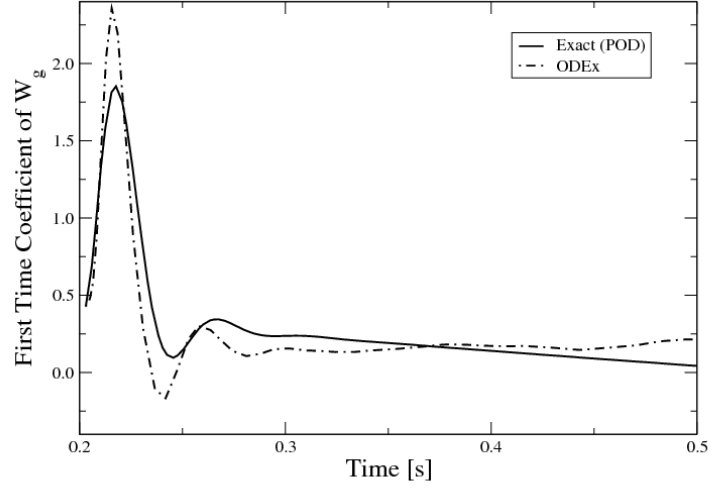
(a)
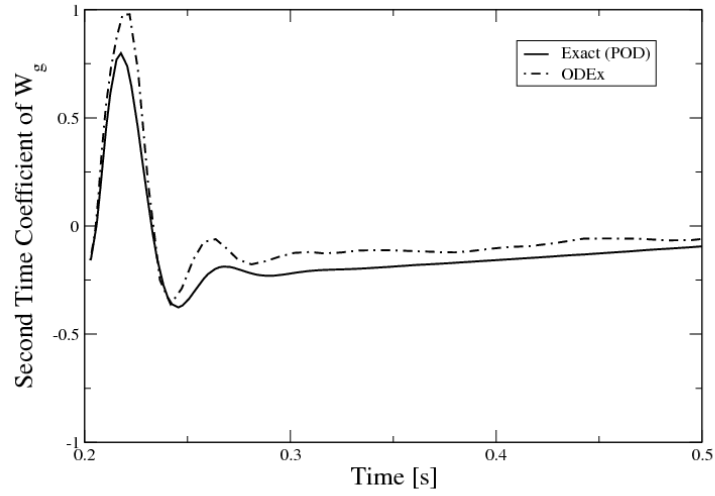
(b)

Figure 15: Time history of the first two time coefficients for z-direction gas velocity in the 3D case: (a) $\alpha_1^{w_g}$, and (b) $\alpha_2^{w_g}$.
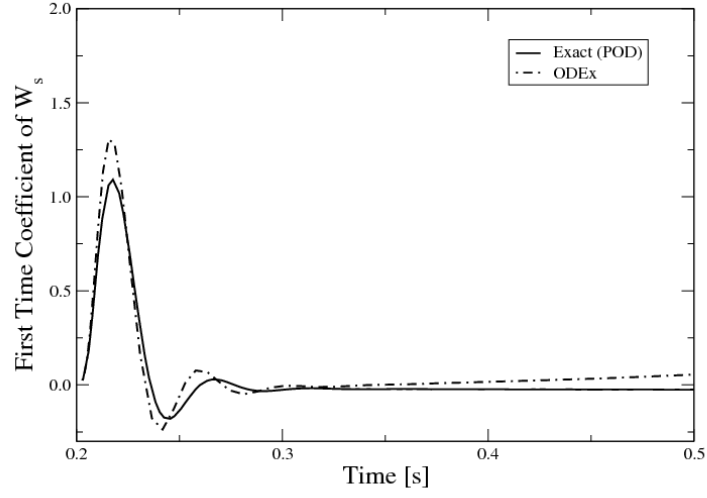
(a)
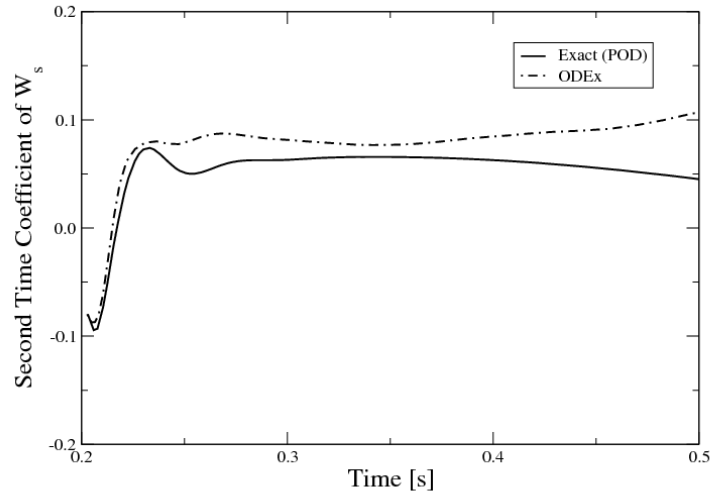
(b)

Figure 16: Time history of the first two time coefficients for z-direction solids velocity in the 3D case: (a) $\alpha_1^{w_s}$, and (b) $\alpha_2^{w_s}$.

Table III: Isothermal case parameters.

Parameter	Description	Value
x_{length}	Length of the domain in x-direction	25.4 cm
y_{length}	Length of the domain in y-direction	76.5 cm
$imax$	Number of cells in x-direction	108
$jmax$	Number of cells in y-direction	124
v_1	Jet gas inflow velocity	12.6 cm/s
v_2	Distributed gas inflow velocity	1.0 cm/s
p_s	Static gas pressure at outlet	1.01×10^6 g/(cm s^2)
T_{g0}	Gas temperature	297 K
μ_{g0}	Gas viscosity	1.8×10^{-4} g/(cm s)
t_{start}	Start time	0 s
t_{stop}	Stop time	1 s
$\triangle t$	Initial time step	1.0×10^{-4} s
ρ_{so}	Constant solids density	1.0 g/cm^3
D_p	Solids particle diameter	0.5 mm
h_{s0}	Initial packed bed height	38.25 cm
ϵ_g^*	Packed bed void fraction	0.40

Table IV: Number of modes used in the three-dimensional case.

Field variable	Symbol	No. of modes
Gas pressure	m^{p_g}	2
Void fraction	m^{ϵ_g}	7
u gas velocity	m^{u_g}	2
v gas velocity	m^{v_g}	6
w gas velocity	m^{w_g}	2
u solids velocity	m^{u_s}	9
v solids velocity	m^{v_s}	9
w solids velocity	m^{w_s}	9

CHAPTER IV

MOVING DISCONTINUITIES*

Flows with moving discontinuities cause the current POD approximation to produce poor results. In the usual POD method, the time average is subtracted from the data prior to computing the POD modes. This average mode captures any stationary discontinuities in the data. Moving discontinuities, however, are approximated as the superposition of modes of increasing frequency. This approximation is susceptible to dispersion errors that result in the well known Gibbs phenomenon of non-physical oscillations about the discontinuity.

Two common examples of moving discontinuities in flows that are challenging for POD-based ROMs are: (i) moving shocks and (ii) gas bubbles in multiphase flow. For flows with moving shocks, a solution of domain decomposition was proposed where the FOM was used in regions where a shock is possible and a ROM was used everywhere else [48].

In multiphase flow, gas bubbles represent moving discontinuities and the resulting oscillations can cause non-physical values for the void fraction. These spurious values for the void fraction can then cascade through the other field variables and lead to divergence of the computation. In multiphase flow, bubbles can exist in most of the physical domain, rendering the domain decomposition approach impractical. The work presented herein seeks to augment the POD bases with non-orthogonal modes that capture the moving discontinuities exactly and prevent the Gibbs phenomenon from occurring.

———————

First, bubbling multiphase flow is discussed and examples are presented showing the problems that ROMs encounter with moving discontinuities. Then, point mode POD is described and tested for an isothermal multiphase flow example. The next section is devoted to developing a very simple model problem for testing augmentation methods. A FOM is derived for the simple model problem, followed by a POD-A ROM. Next the method by which the reference solution was computed is described and the form of the discontinuity modes is discussed. Finally, results showing the effectiveness of the augmentation process are presented.

A. Bubbling Multiphase Flow

Moving discontinuities in the void fraction occur at the boundary of bubbles in multiphase flow. When the void fraction is approximated with POD, unphysical oscillations appear in the reconstruction near the bubble boundaries. These oscillations result in void fractions greater than one, which cause the algorithm implemented in ODEx to diverge.

To see this in practice, consider the test case described in Table V with the geometry shown in Fig. 1. Snapshots were captured every 0.0025 seconds, starting at 1.0 s and ending at 2.0 s. This time domain was selected to capture a flow that was fully developed.

Once the snapshots were used to compute the POD basis functions, shown in Figs. 17 and 18, the snapshots were then projected directly onto the basis functions to compute the best-case values for the time coefficients. These 'exact' time coefficients were then used to reconstruct the solution at $t = 1.5$s. Figures 19 and 20 show this reconstruction for a range of total modes. Note that unphysical values for void fraction, greater than one or less than the minimum packed bed fraction of 0.4, are

Table V: Parameters of a bubbling fluidized bed [69, p. 156].

Parameter	Description	Units	
x_{length}	Length of the domain in x-direction	cm	40.0
y_{length}	Length of the domain in y-direction	cm	76.5
i_{max}	Number of cells in x-direction	-	108
j_{max}	Number of cells in y-direction	-	124
v_1, v_2	Gas inflow velocities	cm/s	577.0, 53.6
p_{g_s}	Static pressure at outlet	g/cm/s^2	$1.01e^6$
T_{g0}	Gas temperature	K	297
μ_{g0}	Gas viscosity	g/cm/s	$1.8e^{-4}$
t_{start}	Start time	s	0.0
t_{stop}	Stop time	s	5.0
ρ_s	Particle density	g/cm^3	2.42
D_p	Particle diameter	cm	0.08
h_{s0}	Initial height of packed bed	cm	29.2
ϵ_g^*	Initial void fraction of packed bed	-	0.4

shown as gaps, or white spaces, in the contour plot data.

It is clear that it is possible to capture almost all of the structures in the flow using only 8 modes. The main effect of continuing to add modes is to eliminate the unphysical regions of the flow. However, even with 40 modes, unphysical regions still exist in the reconstruction. It seems likely that a more efficient means of capturing moving discontinuities exists.

When the POD basis functions were computed, it was noticed that certain basis
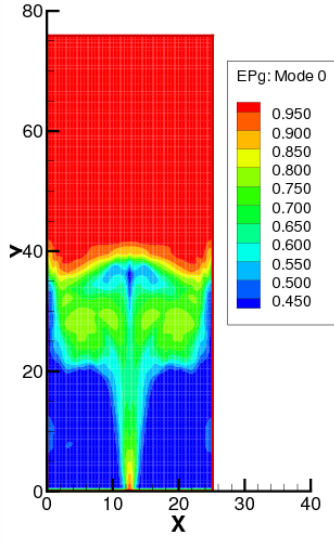
(a)
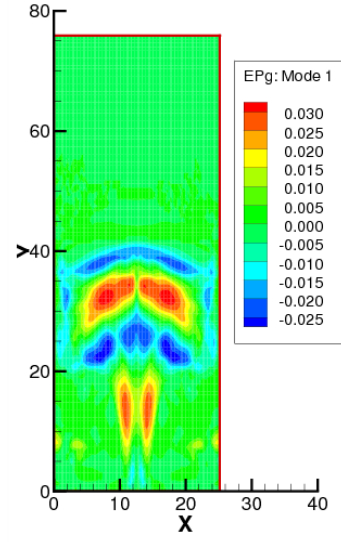
(b)
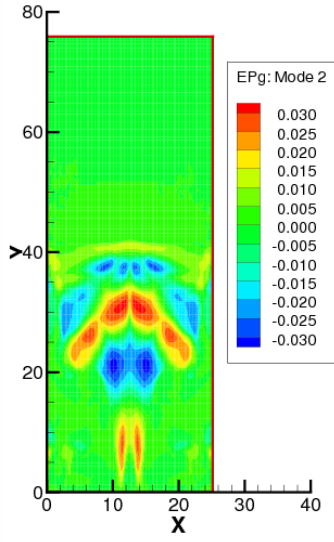
(c)
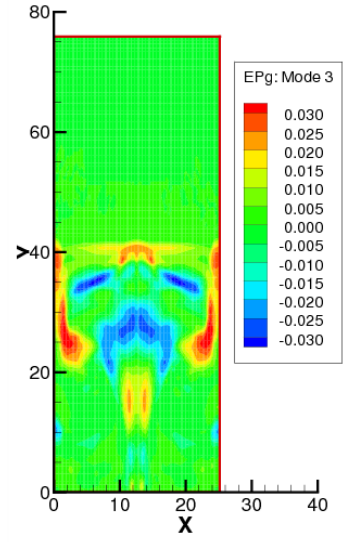
(d)

Figure 17: Contour plots of void fraction basis functions: (a) $\varphi_0^{\epsilon_g}$, (b) $\varphi_1^{\epsilon_g}$, (c) $\varphi_2^{\epsilon_g}$, (d) $\varphi_3^{\epsilon_g}$.
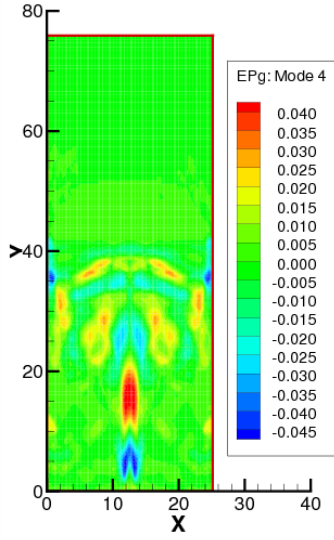
(a)
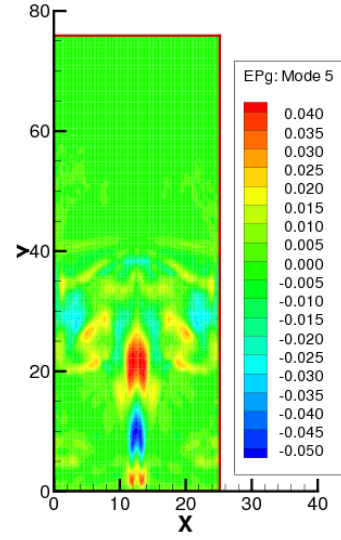
(b)
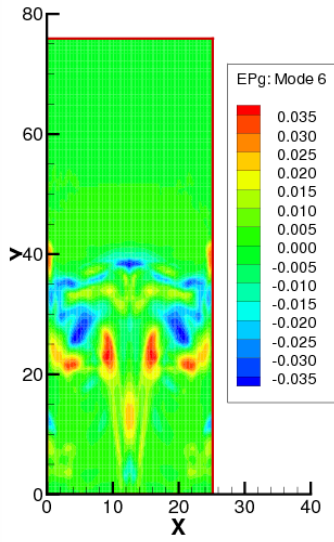
(c)
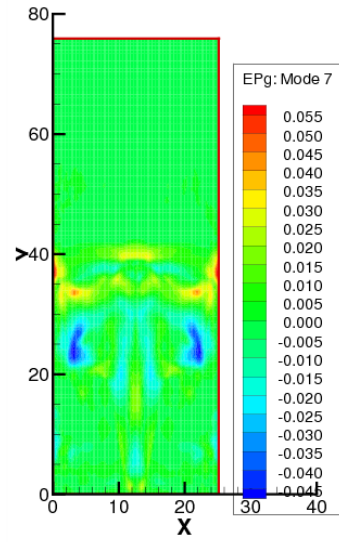
(d)

Figure 18: Contour plots of void fraction basis functions: (a) $\varphi_4^{\epsilon_g}$, (b) $\varphi_5^{\epsilon_g}$, (c) $\varphi_6^{\epsilon_g}$, (d) $\varphi_7^{\epsilon_g}$.
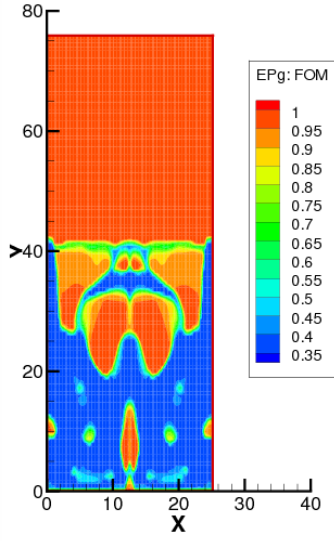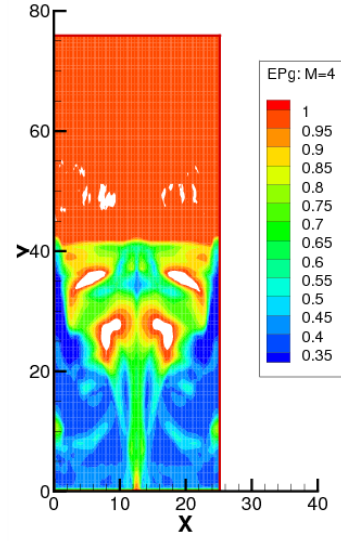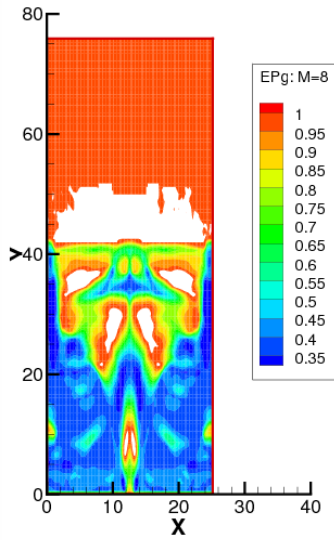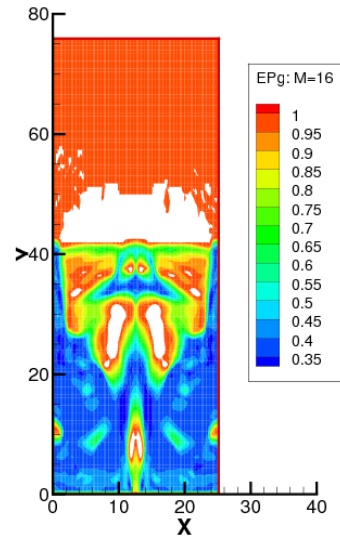
Figure 19: Contour plots of void fraction, ϵ_g, at $t = 1.5$ s for: (a) FOM, (b) POD reconstruction with 4 modes, (c) POD reconstruction with 8 modes, (d) POD reconstruction with 16 modes.
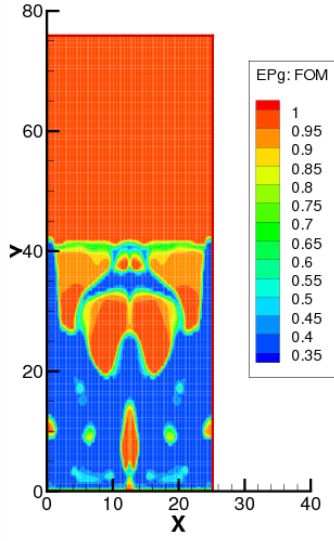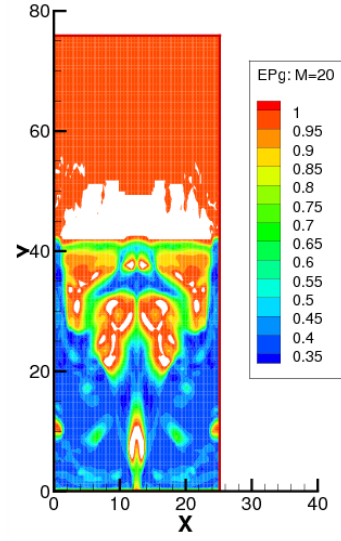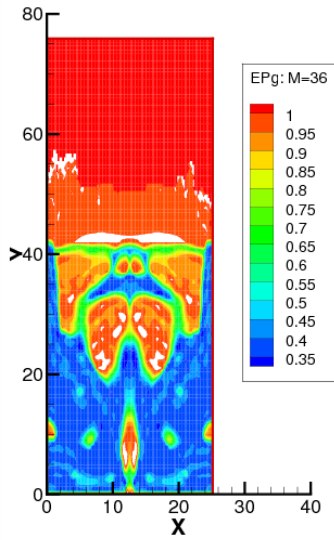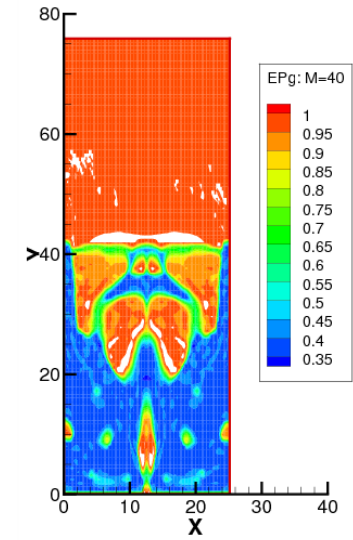
Figure 20: Contour plots of void fraction, ϵ_g, at $t = 1.5$ s for: (a) FOM, (b) POD reconstruction with 20 modes, (c) POD reconstruction with 36 modes, (d) POD reconstruction with 40 modes.

functions for the void fraction resembled a bubble at different locations in the spatial domain. For an example, see $\varphi_4^{\epsilon_g}$ and $\varphi_5^{\epsilon_g}$ in Fig. 18. The time coefficients associated with these basis functions were then computed using the projection method and are shown in Figs. 21 and 22 as a function of time. As these time coefficients were strongly periodic, a Fourier transform was performed to analyze the effect of the basis function in the frequency domain. The resulting frequencies are shown in Figs. 23 and 24.
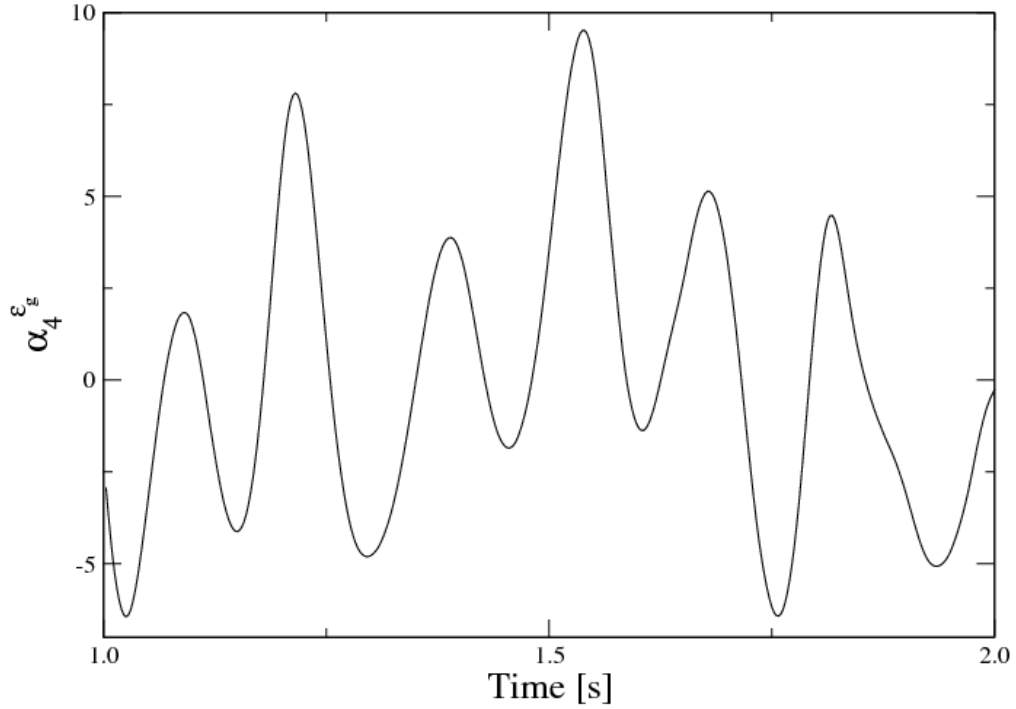
Figure 21: Fourth time coefficient for void fraction.

These results show that the time coefficients associated with basis functions that have bubble-like structures have a dominant frequency of 7 Hz. It is possible to qualitatively determine the bubbling frequency in the fluidized bed by generating an
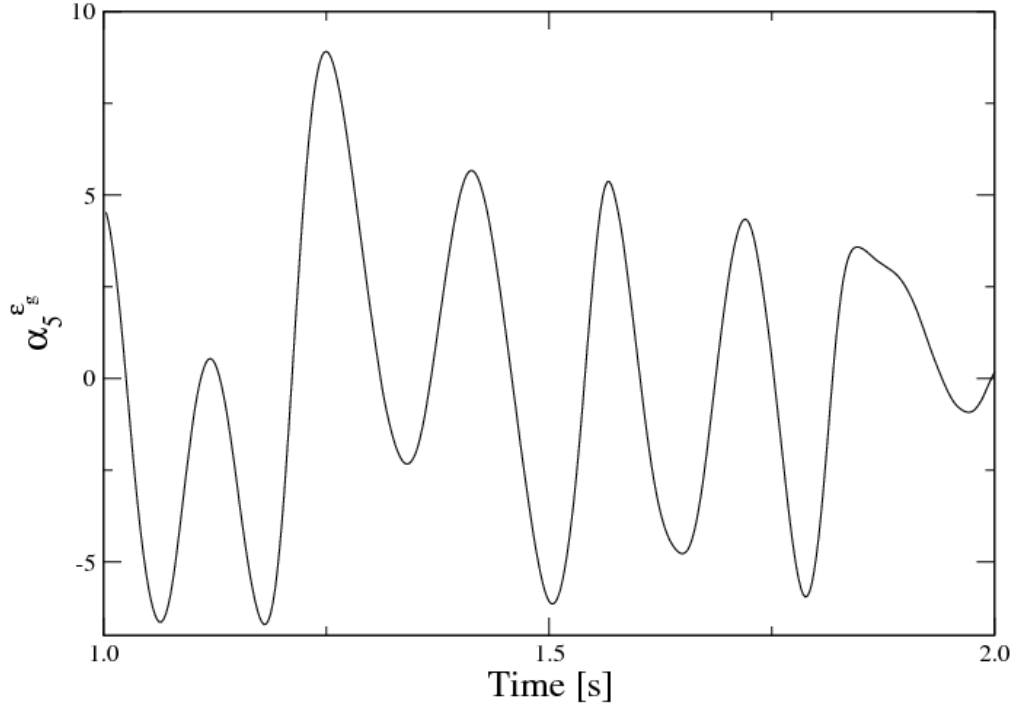
Figure 22: Fifth time coefficient for void fraction.

animation of the flow and counting the number of bubbles that are formed. This method shows that approximately seven bubbles are formed between $t = 1.0$ s and $t = 2.0$ s. Therefore, we conclude that it is possible to determine the bubbling frequency of the flow by applying Fourier analysis to the time coefficients of the POD. Furthermore, for this test case, the fourth and fifth POD modes capture the structure of a bubble.
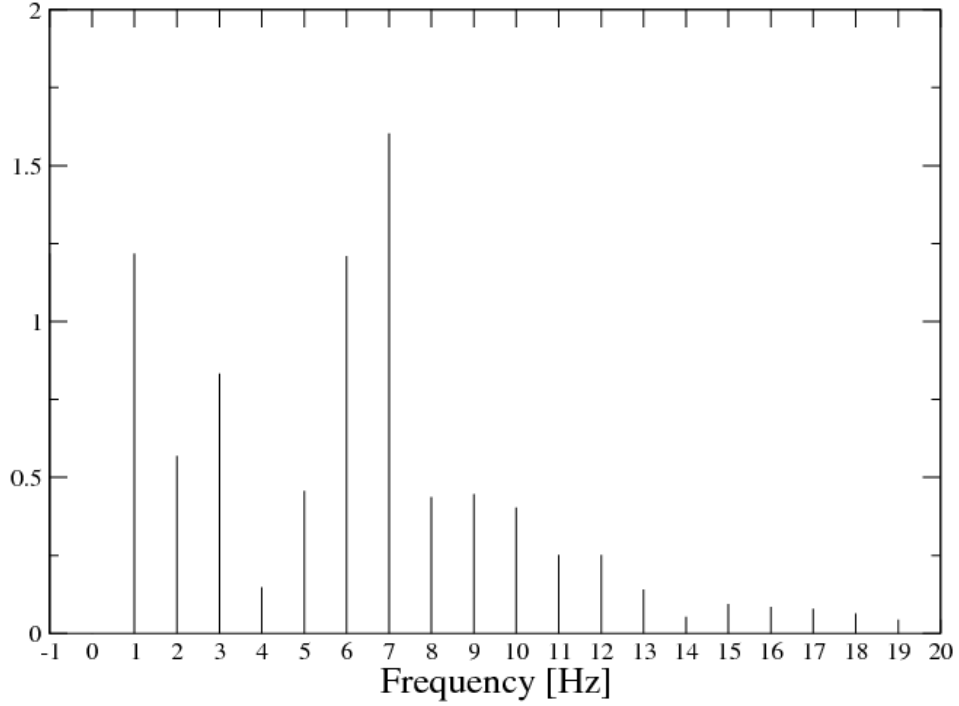
Figure 23: Fourier transform of fourth time coefficient for void fraction.

B.   Point Mode POD

One method for adapting the POD method to solve for moving discontinuities is the point-mode method. In this approach, a collection of point modes are added to the set of POD basis functions. Point modes are defined as having a value of one for a single cell in the computational mesh and are zero everywhere else. If the POD basis functions are then set equal to zero in the cells for which point modes are used, the computational domain will be automatically decomposed into areas where the ROM is solved and areas where the FOM is solved.

To see this, consider the discretized $x-$direction conservation of momentum equa-
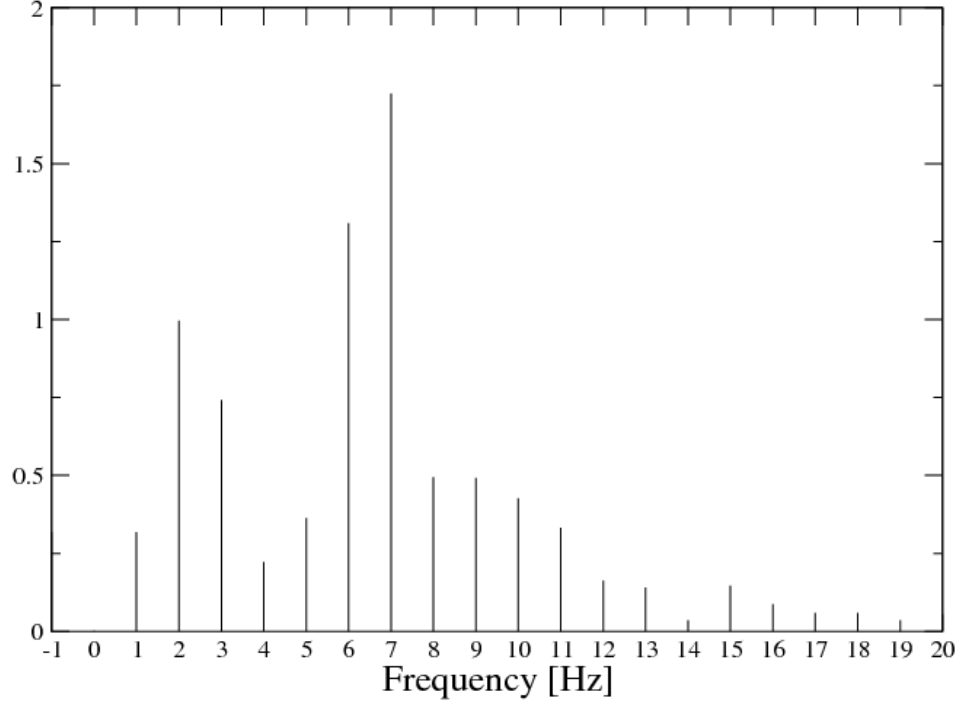
Figure 24: Fourier transform of fifth time coefficient for void fraction.

tion:

$$a_p(u_g)_p + \sum_{nb} a_{nb}(u_g)_{nb} = b_p. \qquad (4.1)$$

The $x-$direction gas velocity can be approximated as

$$u_g(\mathbf{x}, \mathbf{t}) = \varphi_0(\mathbf{x}) + \sum_{j=1}^{n} \varphi_j(\mathbf{x})\alpha_j(\mathbf{t}) + \sum_{j=n+1}^{n+m} \varphi_j^{pm}(\mathbf{x})\alpha_j^{pm}(\mathbf{t}), \qquad (4.2)$$

where φ_j represents a POD basis function and φ_j^{pm} is a point mode. The collection

of POD and point modes is then defined as $\{\psi\}$ where

$$
\psi_j =
\begin{cases}
\varphi_j & 0 \leq j \leq n \\[2ex]
\varphi_j^p & n+1 \leq j \leq n+m
\end{cases}
$$

Note that the POD basis functions that are included in this collection of modes have been masked, that is set to zero, at the spatial points where point modes exist. Substituting (4.2) into (4.1), using a Galerkin projection to project the resulting equation onto $\{\psi\}$, and rearranging, the system of equations to be solved for the time coefficients can be expressed as

$$
A_{ij}\alpha_j = b_i.
$$

Here

$$
A_{ij} = \left(a_p(\psi_j)_p + \sum_{nb} a_{nb}(\psi_j)_{nb}, \psi_i \right)
$$

and

$$
b_i = \left(b_p - \sum_{nb} a_{nb}(\psi_0)_{nb} - a_p(\psi_0)_p, \psi_i \right).
$$

Consider a point mode, φ_j^{pm} defined at a point x_p. This point mode will have a time coefficient, α_j^{pm}. The point mode is orthogonal to every other mode in the set $\{\psi\}$ so, using the x-direction conservation of momentum equation from ODEx, equation for α_j^{pm} is

$$
a_p^{x_p}\alpha^{pm} + \sum_{nb} a_{nb}^{x_p} \sum_{j=0}^{n+m}(\psi_j)_{nb}\alpha_j = b_p^{x_p}
$$

where $a_p^{x_p}, a_{nb}^{x_p}$ and $b_p^{x_p}$ are the coefficients from the discretized momentum equation for the point x_p. Recognizing that

$$
(u_g)_{nb} \approx \sum_{j=0}^{n+m}(\psi_j)_{nb}\alpha_j
$$

regardless of whether the neighbor cell is a point-mode cell or a POD mode cell, then

$$a_p^{x_p} \alpha^{pm} + \sum_{nb} a_{nb}^{x_p} (u_g)_{nb} = b_p^{x_p}.$$

Clearly $\alpha^{pm}(t_k)$ is an approximation for $u_g(x_p, t_k)$.

To demonstrate this, 12 point modes were defined for u_g and used in a test case whose parameters are given in Table III. The geometry for this case is shown in Fig. 1. The point modes were placed in a block, just above the jet inlet. A zoomed view of the average mode, φ_0, is shown in Fig. 25 both before and after masking. Fig. 26 indicates the location of two of the point modes. Finally, Fig. 27 compares the time history of the $x-$direction gas velocity computed with MFIX at these two points to the time history of the time coefficients associated with the two point modes. Good agreement is observed, considering the relatively small variation in the velocity in this simple test case.

The logical next step is to apply this method to the ROM for void fraction. Unfortunately, as ODEx is currently written, the time coefficient associated with a point mode would approximate the void fraction correction, while the ROM is solving for the void fraction itself. An additional modification is needed, either reducing an equation that models the void fraction directly, or developing an algorithm that allows the void fraction to be modeled in some region and the void fraction correction in others. This is a possible topic for future development.

C. Simple Model Problem for Augmentation Methods

The simplest model for a moving discontinuity is the first-order wave, or transport, equation

$$\frac{\partial u}{\partial t} + c \frac{\partial u}{\partial x} = 0. \tag{4.3}$$
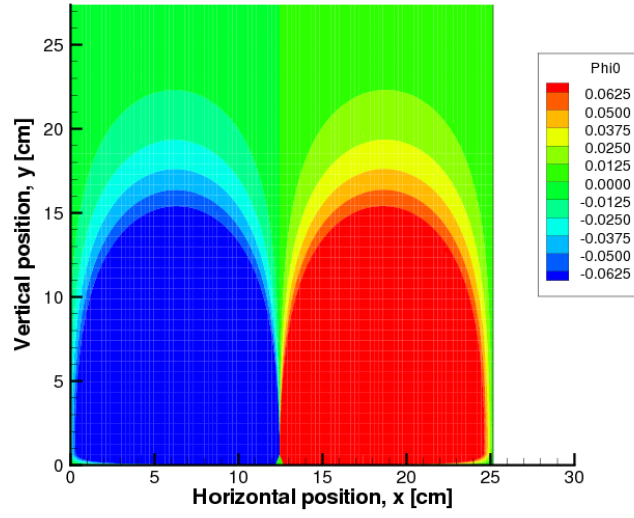
(a)
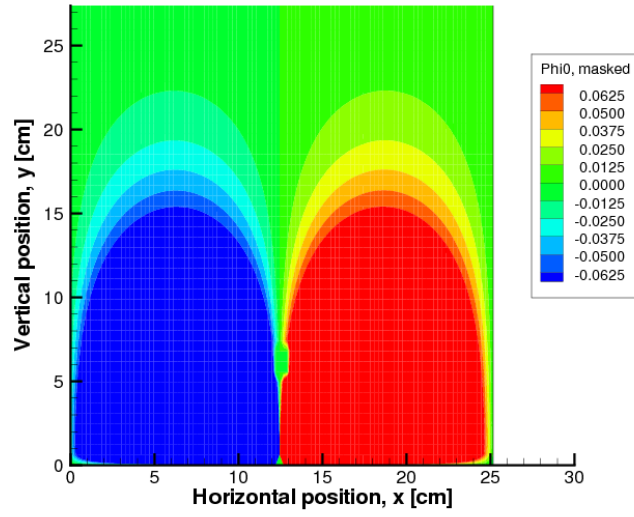
(b)

Figure 25: Contour plots of the average mode for u_g: (a) unmasked, and (b) masked.
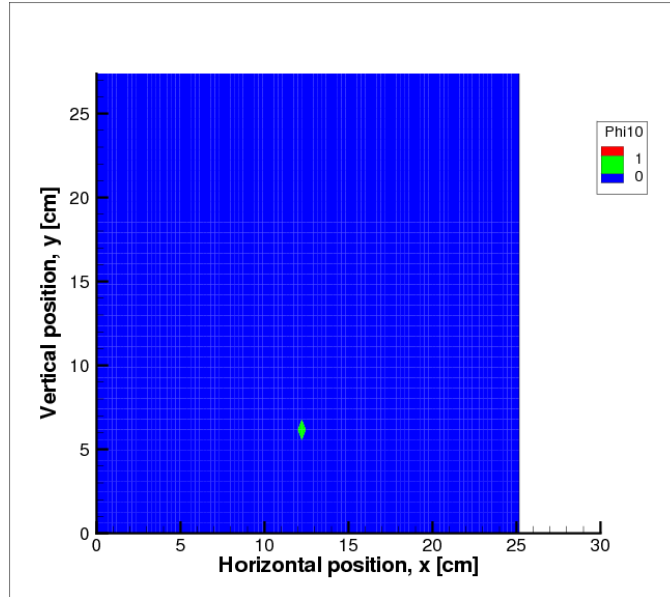
(a)
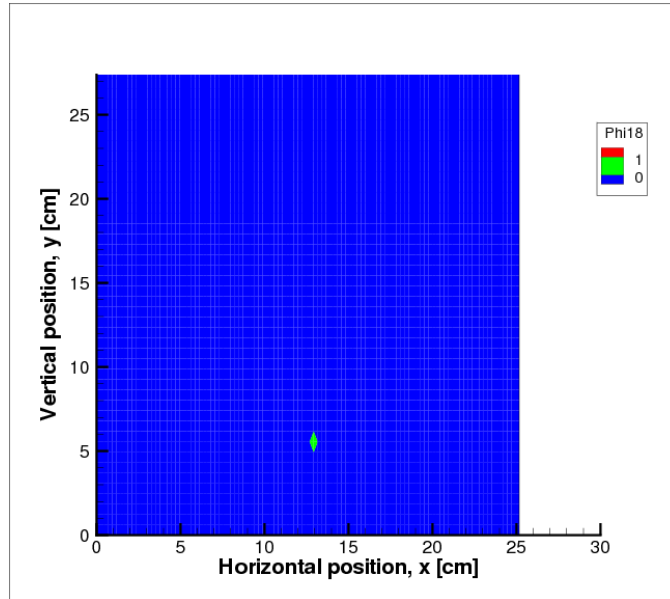
(b)

Figure 26: Contour plots indicating the location of two point modes: (a) ϕ_{10}^{pm}, and (b) ϕ_{18}^{pm}.
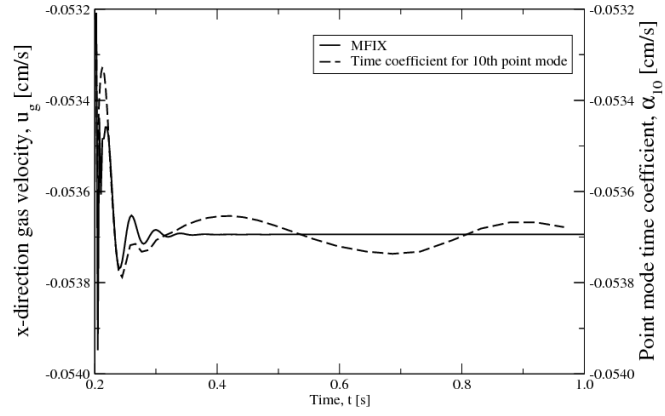
(a)
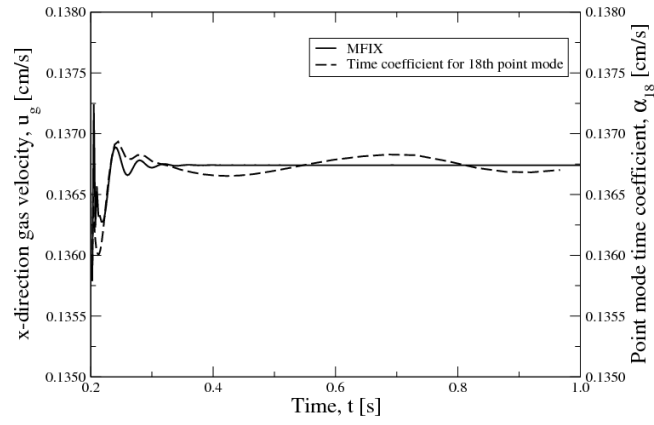
(b)

Figure 27: Time histories for u_g at two points in the flow, compared to: (a) α_{10}^{pm}, and (b) α_{18}^{pm}.

The wave equation models an initial velocity distribution that propagates with velocity c. This model was selected for the ease of predicting the location of the discontinuity, which simplifies the calculation of the discontinuity modes used to augment the POD basis functions in the ROM. At a given time t, the discontinuity location, x_s, is given by

$$x_s = ct + x_{s,0}$$

where $x_{s,0}$ is the initial location of the discontinuity.

The first-order wave equation has a number of exact solutions, but for this problem a simple quadratic polynomial solution given by

$$u(x, t) = a + b(x - ct) + d(x - ct)^2$$

was used. A discontinuity was added to the initial condition at time $t = 0$ through superposition according to

$$u(x, t = 0) = \begin{cases} a + bx + dx^2 + 1 & x \le x_{s,0}. \\ a + bx + dx^2 & x > x_{s,0}. \end{cases}$$

D.   Full-Order Model

Using a first-order forward difference in time and a first-order backward difference in space, (4.3) was discretized as

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} + c\frac{u_i^n - u_{i-1}^n}{\Delta x} = 0 \tag{4.4}$$

where $u_i^{n+1} := u(x_i, t^{n+1})$, $u_i^n := u(x_i, t^n)$, $u_{i-1}^n := u(x_{i-1}, t^n)$, and Δx and Δt were constant over the integration interval. Rearranging (4.4),

$$u_i^{n+1} = u_i^n + \frac{c\Delta t}{\Delta x}(u_i^n - u_{i-1}^n). \tag{4.5}$$

Using (4.5), the solution, u_i^{n+1}, was then computed over the desired time interval and a snapshot was collected at every time step. The time step size was computed using the CFL condition such that

$$\Delta t \leq \frac{\Delta x}{c}.$$

Here, Δx was determined by evenly spacing 100 grid points over the interval $0 \leq x \leq 1$. The propagation speed was $c = 0.5$. One time unit was simulated, resulting in the collection of 50 snapshots.

E. Reduced-Order Model: POD-A

Recall that in (2.1), with the average mode φ_0, a function $u(\mathbf{x}, t)$ can be approximated as,

$$u(\mathbf{x}, t_k) = \varphi_0(\mathbf{x}) + \sum_{j=1}^{m} \alpha_j(t_k)\varphi_j(\mathbf{x}).$$

A series of discontinuity modes, denoted by ψ, can be added to the approximation so that

$$u(\mathbf{x}, t_k) = \varphi_0(\mathbf{x}) + \sum_{j=1}^{m} \alpha_j(t_k)\varphi_j(\mathbf{x}) + \sum_{\ell=1}^{m_a} \beta_\ell(t_k)\psi_\ell(\mathbf{x}, t_k),$$

where β_ℓ are the time coefficients associated with the discontinuity modes ψ_ℓ. Substituting this approximation into (4.3) gives

$$\frac{\partial}{\partial t}\left(\sum_{j=0}^{m} \varphi_j(\mathbf{x})\alpha_j(t_k) + \sum_{\ell=1}^{m_a} \psi_\ell(\mathbf{x}, t_k)\beta_\ell(t_k)\right)$$

$$+ c\frac{\partial}{\partial x}\left(\sum_{j=0}^{m} \varphi_j(\mathbf{x})\alpha_j(t_k) + \sum_{\ell=1}^{m_a} \psi_\ell(\mathbf{x}, t_k)\beta_\ell(t_k)\right) = 0$$

or

$$\sum_{j=1}^{m} \dot{\alpha}_j(t_k)\varphi_j(\mathbf{x}) + \sum_{\ell=1}^{m_a} \dot{\beta}_\ell(t_k)\psi_\ell(\mathbf{x}, t_k) + \sum_{\ell=1}^{m_a} \beta_\ell(t_k)\dot{\psi}_\ell(\mathbf{x}, t_k)$$

$$+ c\sum_{j=0}^{m} \alpha_j(t_k)\varphi_j'(\mathbf{x}) + \sum_{\ell=1}^{m_a} \beta_\ell(t_k)\psi_\ell'(\mathbf{x}, t_k) = 0 \quad (4.6)$$

where the notation $\dot{u} := \frac{\partial u}{\partial t}$ and $u' := \frac{\partial u}{\partial x}$ is used for derivatives. Projecting (4.6) onto the augmented set of basis functions using a Galerkin projection, a system of ODEs is obtained given by

$$[A]\left\{\begin{array}{c}\dot{\alpha}\\\dot{\beta}\end{array}\right\} + [B]\left\{\begin{array}{c}\alpha\\\beta\end{array}\right\} + \{d\} = \{0\}, \qquad (4.7)$$

with $[A] \in \mathbb{R}^{m+m_a} \times \mathbb{R}^{m+m_a}$, $\{\alpha\} \in \mathbb{R}^m$, $\{\beta\} \in \mathbb{R}^{m_a}$ and $\{d\} \in \mathbb{R}^{m+m_a}$. The elements of the $[A]$ and $[B]$ matrices and the $\{d\}$ vector are

$$A_{ij} = \begin{cases} (\varphi_j, \varphi_i) = \delta_{ji} & i = 1, \ldots, m & j = 1, \ldots, m \\ (\varphi_j, \psi_i) & i = m+1, \ldots, m+m_a & j = 1, \ldots, m \\ (\psi_j, \varphi_i) & i = 1, \ldots, m & j = m+1, \ldots, m+m_a \\ (\psi_j, \psi_i) & i = m+1, \ldots, m+m_a & j = m+1, \ldots, m+m_a \end{cases}$$

$$B_{ij} = \begin{cases} c(\varphi_j', \varphi_i) & i = 1, \ldots, m & j = 1, \ldots, m \\ c(\varphi_j', \psi_i) & i = m+1, \ldots, m+m_a & j = 1, \ldots, m \\ (\dot{\psi}_j + c\psi_j', \varphi_i) & i = 1, \ldots, m & j = m+1, \ldots, m+m_a \\ (\dot{\psi}_j + c\psi_j', \psi_i) & i = m+1, \ldots, m+m_a & j = m+1, \ldots, m+m_a \end{cases}$$

and

$$d_i = \begin{cases} c(\varphi_0, \varphi_i) & i = 1, \ldots, m \\ c(\varphi_0, \psi_i) & i = m+1, \ldots, m+m_a \end{cases}$$

Note that the discontinuity modes, ψ_ℓ, are defined as having unit value on one side of a discontinuity and are zero on the other side. The modes are therefore discontinuous and the spatial derivative, ψ'_ℓ, is the Dirac measure, $\delta(x)$, centered at the shock location. Then for $j = m + 1, \ldots, m + m_a$, by definition

$$(c\psi'_j, \varphi_i) = (c \cdot \delta(x_{s_j}), \varphi_i) = c\varphi_i(x_{s_j}) \quad i = 1, \ldots, m$$
$$(c\psi'_j, \psi_i) = (c \cdot \delta(x_{s_j}), \psi_i) = c\psi_i(x_{s_j}) \quad i = m + 1, \ldots, m + m_a$$

where x_{s_j} is the location of the discontinuity in the jth discontinuity mode. Therefore,

$$B_{ij} = \begin{cases} c(\varphi'_j, \varphi_i) & i = 1, \ldots, m & j = 1, \ldots, m \\ c(\varphi'_j, \psi_i) & i = m + 1, \ldots, m + m_a & j = 1, \ldots, m \\ (\dot{\psi}_j, \varphi_i) + c\varphi_i(x_{s_j}) & i = 1, \ldots, m & j = m + 1, \ldots, m + m_a \\ (\dot{\psi}_j, \psi_i) + c\psi_i(x_{s_j}) & i = m + 1, \ldots, m + m_a & j = m + 1, \ldots, m + m_a \end{cases}$$

Note that for the simple test case considered, only one discontinuity mode was used, i.e., $m_a = 1$. An adaptive Runge-Kutta-Fehlberg method [70] was then used to solve the system of ODEs given by (4.7).

F.   Reference Solution

A set of time coefficients were calculated directly by projecting the snapshots onto the basis functions [10]. This gave the best case solution for the flow at reference conditions. The POD approximation

$$u_a(\mathbf{x}, t_k) = \sum_{j=0}^{m} \alpha_j(t_k)\varphi_j(\mathbf{x}) + \sum_{\ell=1}^{m_a} \beta_\ell(t_k)\psi_\ell(\mathbf{x}, t_k)$$

was then projected onto the augmented set of basis functions using a Galerkin projection such that

$$(u_a - \varphi_0, \varphi_i) = (\sum_{j=1}^{m} \alpha_j \varphi_j + \sum_{\ell=1}^{m_a} \beta_\ell \psi_\ell, \varphi_i) \quad i = 1, \ldots, m$$

$$(u_a - \varphi_0, \psi_i) = (\sum_{j=1}^{m} \alpha_j \varphi_j + \sum_{\ell=1}^{m_a} \beta_\ell \psi_\ell, \psi_i) \quad i = m+1, \ldots, m+m_a.$$

It is possible for the discontinuity modes to not be orthogonal to the POD bases. As a result the system does not decouple and the time coefficients are found by solving the system

$$[A_E] \begin{Bmatrix} \alpha \\ \beta \end{Bmatrix} = \{b_E\}$$

where, using i and j to denote the modes,

$$A_{Eij} = \begin{cases} (\varphi_j, \varphi_i) = \delta_{ji} & i = 1, \ldots, m & j = 1, \ldots, m \\ (\varphi_j, \psi_i) & i = m+1, \ldots, m+m_a & j = 1, \ldots, m \\ (\psi_j, \varphi_i) & i = 1, \ldots, m & j = m+1, \ldots, m+m_a \\ (\psi_j, \psi_i) & i = m+1, \ldots, m+m_a & j = m+1, \ldots, m+m_a \end{cases}$$

and

$$b_{Ei} = \begin{cases} (u_a - \varphi_0, \varphi_i) & i = 1, \ldots, m \\ (u_a - \varphi_0, \psi_i) & i = m+1, \ldots, m_a. \end{cases}$$

G. Discontinuity Modes

Up to this point, very little has been said about the structure of the discontinuity modes. Note that the POD basis functions do not span the entire space because the set of basis functions was truncated and the highest-order modes discarded. The discontinuity modes must be linearly independent of the POD basis functions and it was therefore necessary to test the augmented set of basis functions to check whether the discontinuity modes, $\{\psi\}$ could be expressed as a linear combination of $\varphi_i, i \in$

$[0, m]$.

To ensure this linear independence, the discontinuity modes were appended to the matrix of basis functions at every time step and the singular value decomposition was computed. If the smallest singular value of this matrix was less than an error tolerance, the corresponding discontinuity mode was not linearly independent and was therefore not needed in the calculation.

For simplicity, herein each discontinuity mode was a simple step function with a value of one upstream of the discontinuity and zero downstream of the discontinuity. For the simple test case considered in this paper, only one discontinuity was present at any time and its location was always known, so only one discontinuity mode was needed. In practice, a number of discontinuity modes could have been used, each representing a possible location for the discontinuity. Since each discontinuity mode in this collection was constant in time, $\dot{\psi}_\ell$ was assumed to be zero.

H.   Results

The velocity profile for the first-order wave equation case is shown in Fig. 28 at $t = 1.0$ for the FOM, the non-augmented reduced-order model and the augmented reduced-order model. The non-augmented ROM used 20 modes while the augmented ROM used only two modes. In the augmented case, the discontinuity was removed prior to computing the POD modes, as it is captured by the discontinuity mode. Figure 29 shows the time average and first four POD modes used by the non-augmented ROM. The non-augmented ROM shows the Gibbs phenomenon while the augmented ROM does not have oscillations present.

Figs. 30 and 31 show the first two time coefficients, respectively, computed using the ROM with a discontinuity mode, compared to the results obtained by projecting
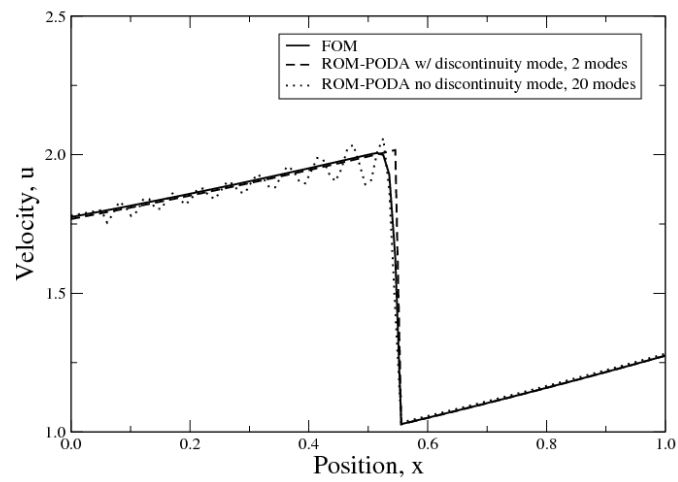
Figure 28: Velocity profile for the one-dimensional wave equation model problem at $t = 1.0$ for full-order model, reduced-order model with 20 modes and no discontinuity mode, and reduced-order model with 2 modes, including a discontinuity mode.
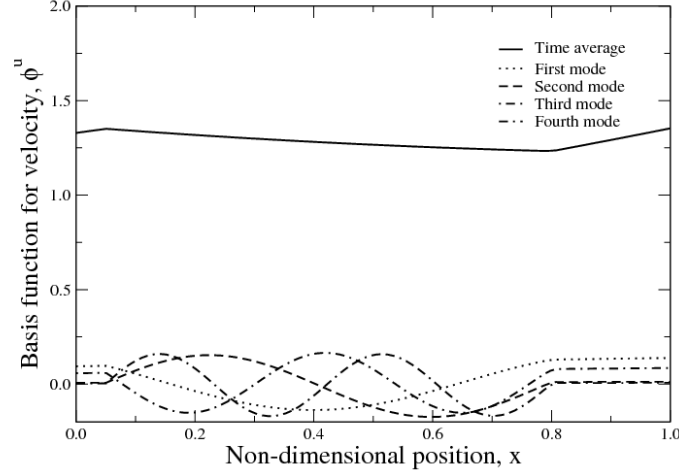
Figure 29: Time average and first four POD modes for the first order wave equation model problem.

the snapshots directly onto the basis functions. Note that for this problem, the second mode time coefficient is the discontinuity mode time coefficient.

The computed time coefficients show excellent agreement with the reference solution. This is further reinforced by the lack of Gibbs oscillations in the solution of the augmented POD method. The additional computations created by the non-orthogonality of the modes are more than accounted for by the reduction in modes from 20 to 2 when the augmented ROM is compared to the traditional ROM. As an additional benefit, the solution computed using the augmented ROM is not degraded by numerical dissipation around the discontinuity as the FOM is.
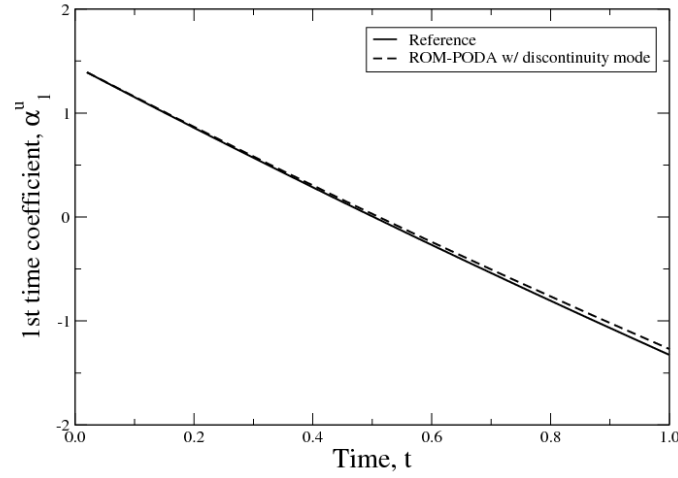
Figure 30: First mode time coefficient for the one-dimensional wave equation model problem computed using the augmented ROM and compared to a reference computed by projecting the snapshots onto the basis functions.
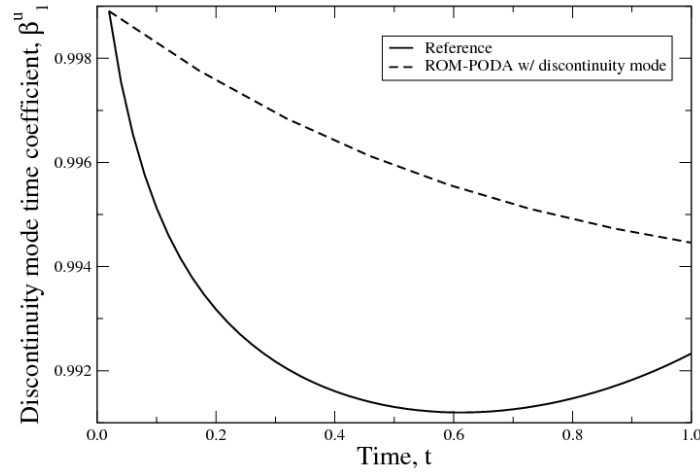
Figure 31: Time coefficient for discontinuity mode for the one-dimensional wave equation model problem computed using the augmented ROM and compared to a reference computed by projecting the snapshots onto the basis functions.

CHAPTER V

NOZZLE FLOW

In preparation for implementation of a ROM for a Navier-Stokes solver, an intermediate ROM was created for the quasi-2D Euler equations. This ROM was used to evaluate two approaches, POD-A and POD-D, to determine which approach was best. The next section describes the quasi-2D Euler equations for nozzle flow. After that, the FOM used to solve this model is discussed. Then, the approaches for the POD-A and POD-D ROMs are described and the ROMs are derived. Results are presented comparing the two approaches. Finally, the stability for the two approaches is discussed are efforts are made to increase the stability of the model.

A.   Physical Model

A quasi-2D inviscid model was used to compare the POD-A and POD-D approaches. The test case used was a subsonic flow through a convergent-divergent nozzle. A full-order model was constructed using a MacCormack scheme as described by Anderson [71]. The conservation of mass, momentum, and energy equations, expressed in non-dimensional, non-conservative, differential form, were

$$\frac{\partial \rho}{\partial t} = -\rho \frac{\partial V}{\partial x} - \rho V \frac{\partial (\ln A)}{\partial x} - V \frac{\partial \rho}{\partial x}, \tag{5.1}$$

$$\frac{\partial V}{\partial t} = -V \frac{\partial V}{\partial x} - \frac{1}{\gamma} \Big(\frac{\partial T}{\partial x} + \frac{T}{\rho} \frac{\partial \rho}{\partial x} \Big), \tag{5.2}$$

and

$$\frac{\partial T}{\partial t} = -V \frac{\partial T}{\partial x} - (\gamma - 1)T \Big[\frac{\partial V}{\partial x} + V \frac{\partial (\ln A)}{\partial x} \Big]. \tag{5.3}$$

Here ρ, V, T and A are the density, velocity, temperature and cross-sectional area, respectively.

B.   Full-Order Model

An adaptation of a finite-difference method was used to construct a FOM for quasi-2D nozzle flow. This method was taken from Anderson [71] and was a MacCormack predictor-corrector technique. This method is summarized below.

1. Using first-order forward finite-differences to approximate the spatial derivatives, compute values for $\frac{\partial \rho}{\partial t}$, $\frac{\partial V}{\partial t}$ and $\frac{\partial T}{\partial t}$ based on the field variables at the current time step.

2. Compute predicted values for ρ, V, and T at the next time step according to $\bar{\aleph}_i^{t+\Delta t} = \aleph_i^t + (\frac{\partial \aleph}{\partial t})_i^t \Delta t$, where $\aleph$ is any of the field variables.

3. Using first-order rearward finite differences to approximate the spatial derivatives, compute values for $\frac{\bar{\partial} \rho}{\partial t}$, $\frac{\bar{\partial} V}{\partial t}$ and $\frac{\bar{\partial} T}{\partial t}$ based on the predicted field variables at the next time step.

4. Compute the average time derivative according to $(\frac{\partial \aleph}{\partial t})_{av} = \frac{1}{2}[(\frac{\partial \aleph}{\partial t})_i^t + (\frac{\bar{\partial} \aleph}{\partial t})_i^{t+\Delta t}]$

5. Finally, $\aleph_i^{t+\Delta t} = \aleph_i^t + (\frac{\partial \aleph}{\partial t})_{av} \Delta t$.

The time step, Δt, was selected using a CFL condition to compute the ideal local time step for each node and then using the smallest computed value as the global time step, ensuring time accuracy.

Since discontinuities can cause serious problems in POD-based ROMs, a strictly subsonic test case was used to evaluate the ROMs. Both the inlet and outlet boundary conditions were therefore subsonic. At the inlet a constant density and temperature were assumed and the velocity was extrapolated from inside the domain. At the outlet a pressure is specified. The temperature and velocity are then extrapolated

from the interior and the outlet density is computed using the equation of state. Linear density, temperature and velocity profiles were assumed for initial conditions.

The cross-sectional area of the nozzle was given by

$$A = \begin{cases} 1 + 2.2(x - 1.5)^2 & 0 \leq x \leq 1.5 \\ 1 + 0.2223(x - 1.5)^2 & 1.5 \leq x \leq 3.0 \end{cases}$$

where all quantities are non-dimensional. The back pressure was 93% of the stagnation pressure. The linear initial conditions were

$$\rho = 1.0 - 0.023x,$$

$$T = 1.0 - 0.009333x$$

and

$$V = 0.05 + 0.11x.$$

Approximately 100 s of flow were simulated which was sufficient to reach a steady state.

C. Reduced-Order Model

1. POD-D Method

The state variables are approximated as

$$\rho(\mathbf{x}, t_k) = \varphi_0^\rho(\mathbf{x}) + \sum_{j=1}^{m^\rho} \varphi_j^\rho(\mathbf{x}) \alpha_j^\rho(t_k), \tag{5.4}$$

$$V(\mathbf{x}, t_k) = \varphi_0^V(\mathbf{x}) + \sum_{j=1}^{m^V} \varphi_j^V(\mathbf{x}) \alpha_j^V(t_k), \tag{5.5}$$

and

$$T(\mathbf{x}, t_k) = \varphi_0^T(\mathbf{x}) + \sum_{j=1}^{m^T} \varphi_j^T(\mathbf{x})\alpha_j^T(t_k). \tag{5.6}$$

Here the basis functions, φ^ρ, φ^V, and φ^T are known. The time coefficients are computed using the MacCormack scheme described below.

1. $\alpha_j^\rho(t_n), \alpha_j^V(t_n)$, and $\alpha_j^T(t_n)$ are known.

2. $\rho(x_i, t_n), V(x_i, t_n)$ and $T(x_i, t_n)$ are reconstructed from (5.4)-(5.6).

3. An initial guess for the time derivative for each time coefficient is computed according to $\left(\frac{\partial \alpha_k^\aleph}{\partial t}\right)^{t_n} = \left(\left(\frac{\partial \aleph}{\partial t}\right)_i^{t_n}, \phi_k^\aleph(x_i)\right)$ where $\aleph$ is any field variable and $\frac{\partial \aleph}{\partial t}$ is computed using (5.1)-(5.3).

4. $(\bar{\alpha}_k^\aleph)^{t_n+\Delta t} = (\alpha_k^\aleph)^{t_n} + \Delta t \left(\frac{\partial \alpha_k^\aleph}{\partial t}\right)^{t_n}$.

5. $\bar{\rho}(x_i, t_n+\Delta t), \bar{V}(x_i, t_n+\Delta t)$ and $\bar{T}(x_i, t_n+\Delta t)$ are reconstructed from (5.4)-(5.6) using $(\bar{\alpha}_k^\aleph)^{t_n+\Delta t}$ in place of $\alpha_j^\aleph(t_n)$.

6. An updated guess for the time derivative for each time coefficient is computed according to $\left(\frac{\partial \bar{\alpha}_k^\aleph}{\partial t}\right)^{t_n+\Delta t} = \left(\left(\frac{\partial \bar{\aleph}}{\partial t}\right)_i^{t_n+\Delta t}, \phi_k^\aleph(x_i)\right)$.

7. The average time derivative for each time coefficient is computed according to $\left(\frac{\partial \alpha_k^\aleph}{\partial t}\right)_{av}^{t_n} = \frac{1}{2}\left(\left(\frac{\partial \alpha_k^\aleph}{\partial t}\right)^{t_n} + \left(\frac{\partial \bar{\alpha}_k^\aleph}{\partial t}\right)^{t_n+\Delta t}\right)$.

8. Finally, $\left(\alpha_k^\aleph\right)^{t_n+\Delta t} = \left(\alpha_k^\aleph\right)^{t_n} + \Delta t \left(\frac{\partial \alpha_k^\aleph}{\partial t}\right)_{av}^{t_n}$.

2.   POD-A Method

In the POD-A Method, (5.4)-(5.6) are substituted into (5.1)-(5.3) and the right hand sides of the resulting equations are then projected onto $\{\phi^\rho(x)\}, \{\phi^V(x)\}$ and $\{\phi^T(x)\}$,

respectively, to form the nonlinear operator $f(\alpha)$ in the non-linear system of ODEs

$$\dot{\alpha} = f(\alpha) \tag{5.7}$$

where

$$\alpha = \left\{ \begin{array}{c} \alpha^\rho \\ \alpha^V \\ \alpha^T \end{array} \right\}$$

$$f = \left\{ \begin{array}{c} f^\rho \\ f^V \\ f^T \end{array} \right\}$$

$$f_k^\rho = \left(-(\varphi_0^\rho + \sum_{i=1}^{m^\rho} \alpha_i^\rho \varphi_i^\rho)(\frac{d\varphi_0^v}{dx} + \sum_{j=1}^{m^V} \alpha_j^V \frac{d\varphi_j^V}{dx}) - (\varphi_0^\rho + \sum_{i=1}^{m^\rho} \alpha_i^\rho \varphi_i^\rho)(\varphi_0^V + \sum_{i=1}^{m^V} \alpha_i^V \varphi_i^V)\frac{d(\ln A)}{dx} \right.$$

$$\left. - (\varphi_0^V + \sum_{i=1}^{m^V} \alpha_i^V \varphi_i^V)(\frac{d\varphi_0^\rho}{dx} + \sum_{j=1}^{m^\rho} \alpha_j^\rho \frac{d\varphi_j^\rho}{dx}), \varphi_k^\rho \right),$$

$$f_k^V = \left(-(\varphi_0^V + \sum_{i=1}^{m^V} \alpha_i^V \varphi_i^V)(\frac{d\varphi_0^v}{dx} + \sum_{j=1}^{m^V} \alpha_j^V \frac{d\varphi_j^V}{dx}) - \frac{1}{\gamma}(\frac{d\varphi_0^T}{dx} + \sum_{i=1}^{m^T} \alpha_i^T \frac{d\varphi_i^T}{dx} \right.$$

$$\left. + \frac{\varphi_0^T + \sum_{i=1}^{m^T} \alpha_i^T \varphi_i^T}{\varphi_0^\rho + \sum_{i=1}^{m^\rho} \alpha_i^\rho \varphi_i^\rho}(\frac{d\varphi_0^\rho}{dx} + \sum_{j=1}^{m^\rho} \alpha_j^\rho \frac{d\varphi_j^\rho}{dx})), \varphi_k^V \right),$$

and

$$f_k^T = \left(-(\varphi_0^V + \sum_{i=1}^{m^V} \alpha_i^V \varphi_i^V)(\frac{d\varphi_0^T}{dx} + \sum_{i=1}^{m^T} \alpha_i^T \frac{d\varphi_i^T}{dx}) \right.$$

$$\left. - (\gamma - 1)(\varphi_0^T + \sum_{i=1}^{m^T} \alpha_i^T \varphi_i^T)\left[(\frac{d\varphi_0^v}{dx} + \sum_{j=1}^{m^V} \alpha_j^V \frac{d\varphi_j^V}{dx}) + (\varphi_0^V + \sum_{i=1}^{m^V} \alpha_i^V \varphi_i^V)\frac{d(\ln A)}{dx}\right], \varphi_k^T \right).$$

The system (5.7) is then integrated using a 4th-order Runge-Kutta-Fehlberg routine [70].

D. Stability

The FOM was used to simulate about 100 seconds of flow. During this interval, 1000 snapshots were collected and PODDEC was used to compute the POD basis functions. Figs. 32-34 show the time average and first four basis functions for ρ, V, and T, respectively.
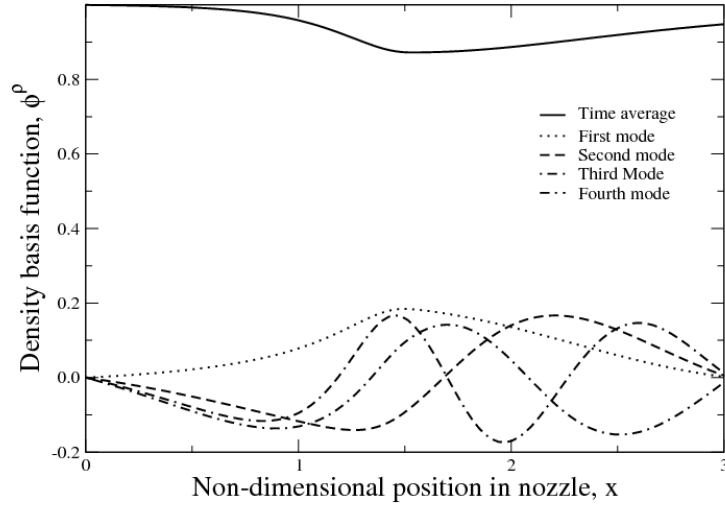
Figure 32: Time average and first four modes for density in a subsonic nozzle.

The first POD-A based ROM used a Runge-Kutta-Fehlberg integration method, but this implementation proved unstable. Figs. 35-37 show the first time coefficient for density, velocity and temperature, respectively, computed using both the POD-A and POD-D methods for the simplest test case, a quasi-2D subsonic nozzle iterating to
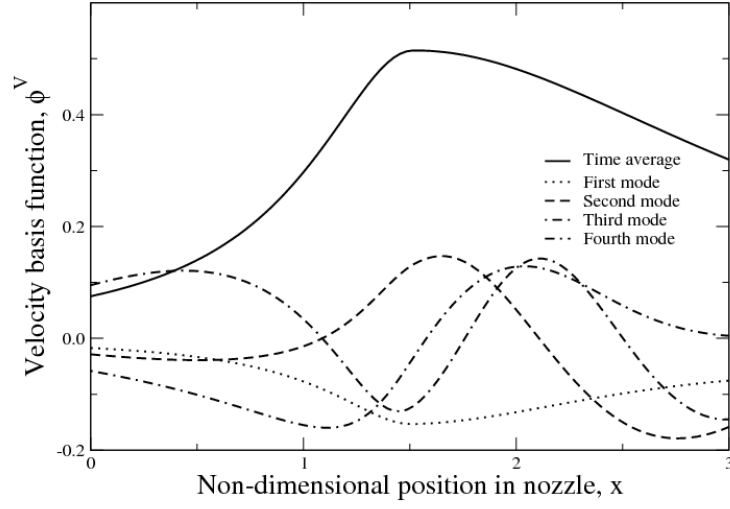
Figure 33: Time average and first four modes for velocity in a subsonic nozzle.
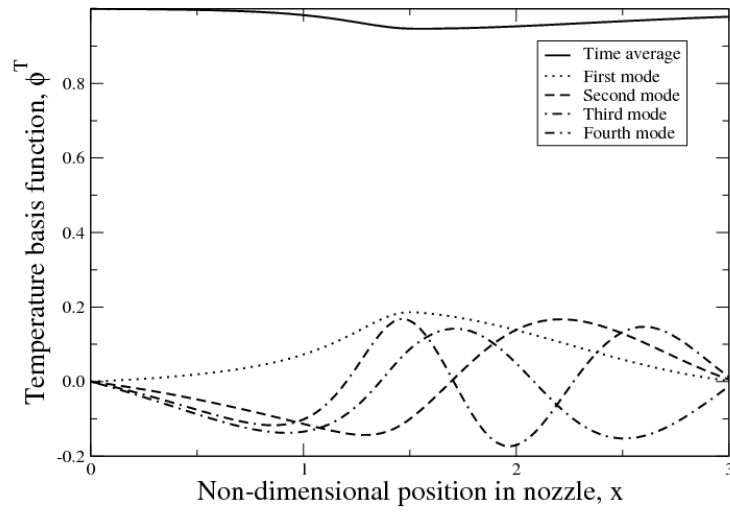
Figure 34: Time average and first four modes for temperature in a subsonic nozzle.

a steady state. Snapshots of the FOM were projected onto the POD basis functions to generate a reference solution. During the computation the POD-A method diverged.
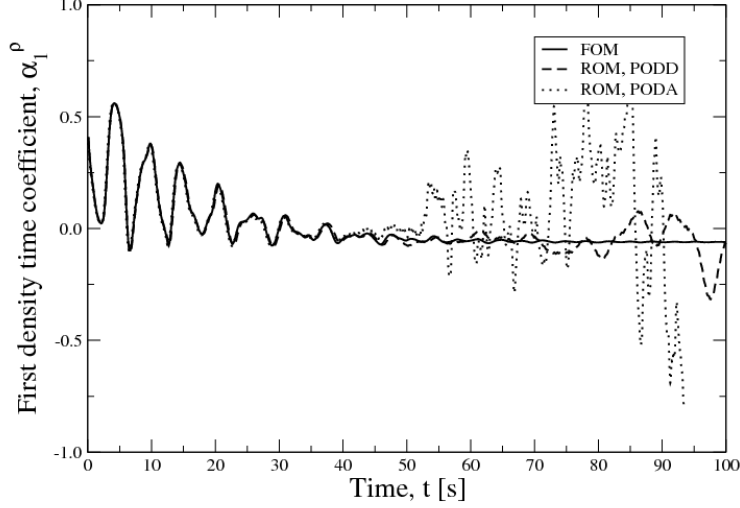
Figure 35: First time coefficient for density in the nozzle case.

The Jacobian of the ODE generated using the POD-A method was computed during integration to give a measure of the stability of the system according to

$$J_{ij} = \frac{\partial f_i}{\partial \alpha_j}.$$

Fig. 38 shows the maximum real component of the eigenvalues of the Jacobian matrix. When the largest real component of the eigenvalues of the Jacobian matrix exceeds one, the system becomes stiff and the calculation is unstable. In order to integrate the stiff system more accurately, several ODE solvers for stiff systems were tested. The best results were obtained using the DVODE routine [72].

This was, unfortunately, only half of the problem. POD-based ROMs also tend
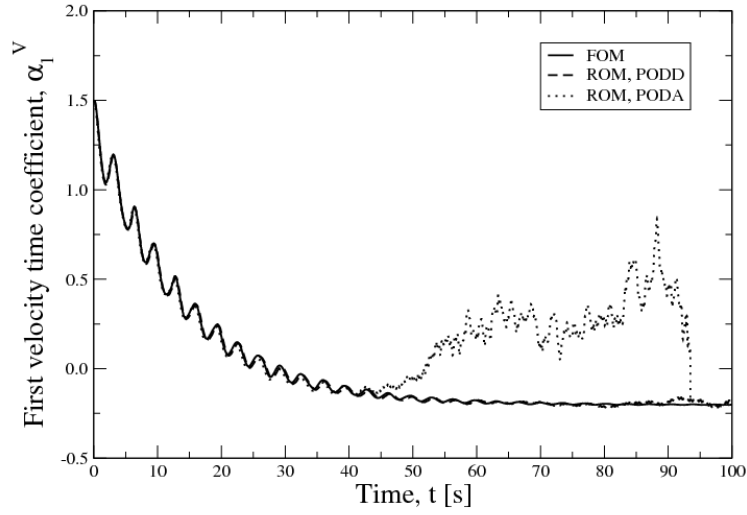
Figure 36: First time coefficient for velocity in the nozzle case.
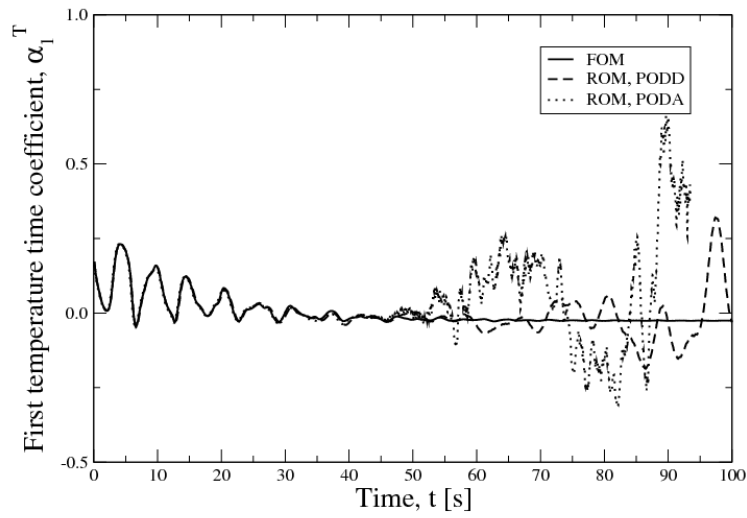
Figure 37: First time coefficient for temperature in the nozzle case.

to become unstable as the time coefficients approach a steady state. A second test case was implemented where an unsteady source term was added to the conservation of energy equation to force an oscillation in the flow. This gave the flow two distinct regimes; a transient regime where the flow is dominated by the transition from the initial condition to a periodic regime and the periodic regime where the flow is dominated by the oscillations in the energy equation. An interesting area for future work lies in properly capturing the transition from the early transience to the periodic flow.
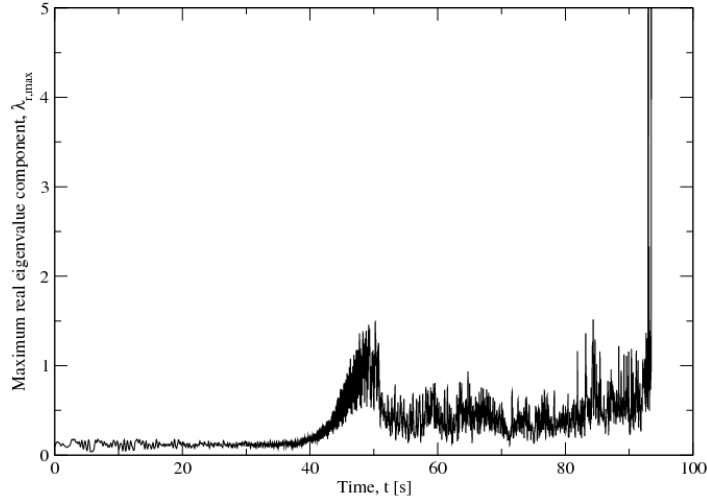
Figure 38: Maximum real component of the eigenvalues of the Jacobian matrix for the quasi-2D Euler equations for a nozzle flow test case.

Figs. 39-41 show the first time coefficients for density, velocity and temperature, respectively, as a result of integrating the governing ODEs for just the transient region using POD basis functions computed from the snapshots of the transient region.

Figs. 42-44 show the first time coefficients for density, velocity and temperature, respectively, as a result of integrating the governing ODEs for just the periodic region.
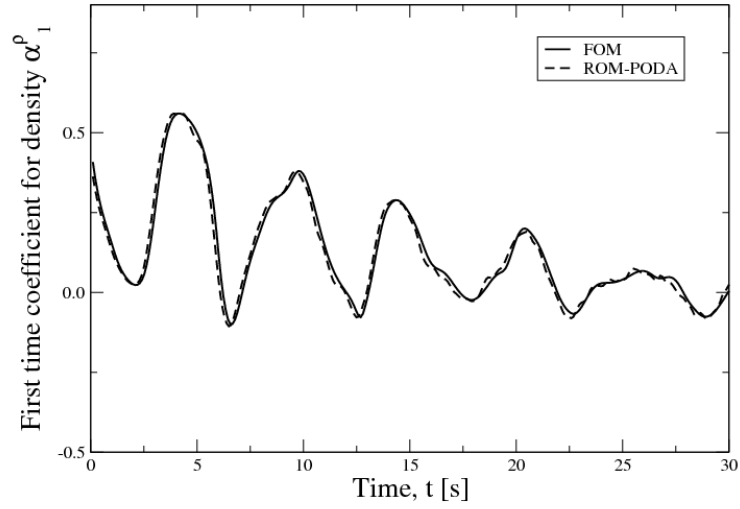
Figure 39: First time coefficient for density in the transient region case.
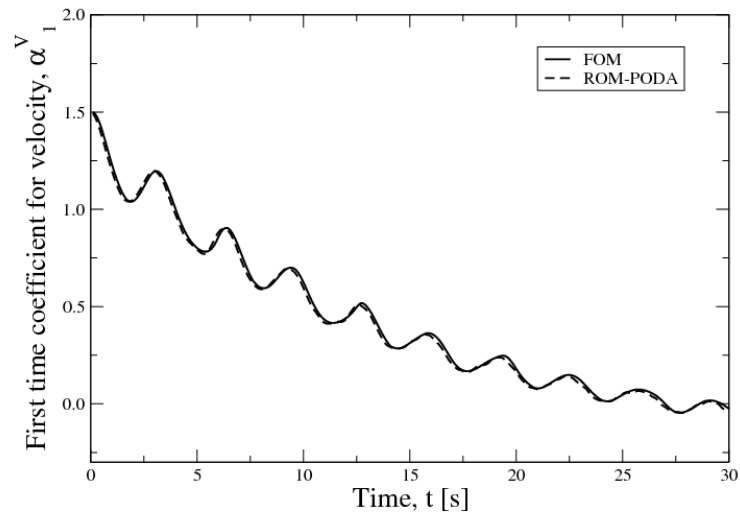
Figure 40: First time coefficient for velocity in the transient region case.
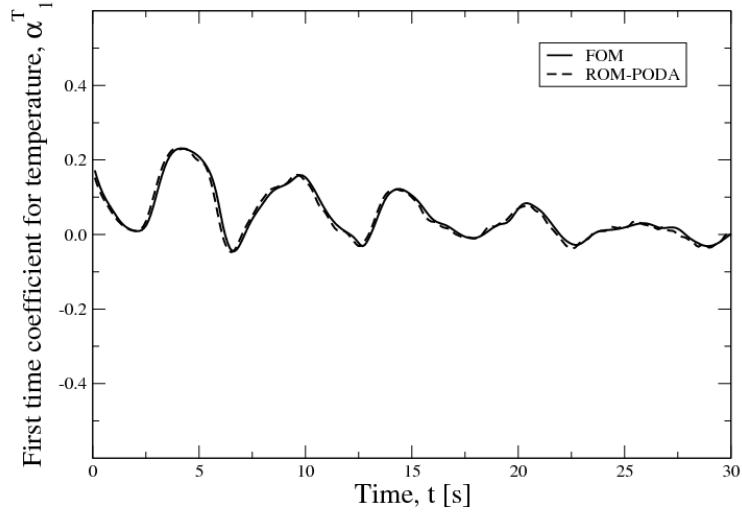
Figure 41: First time coefficient for temperature in the transient region case.

Here all the initial transience has died out.

   The POD-D model required considerably less effort to stabilize and produced a solver that was stable throughout the temporal domain regardless of flow regime. As a result, it was decided that the POD-D method should be used to derive a ROM for use with UNS3D. This process is described in the next chapter.
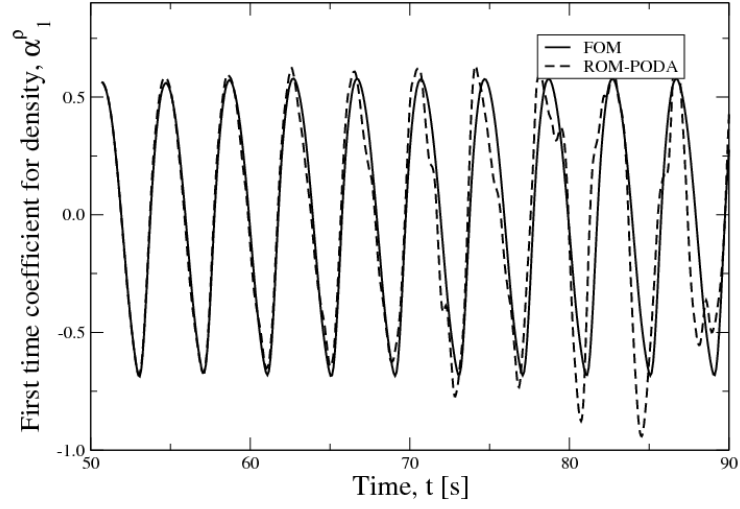
Figure 42: First time coefficient for density in the periodic region case.
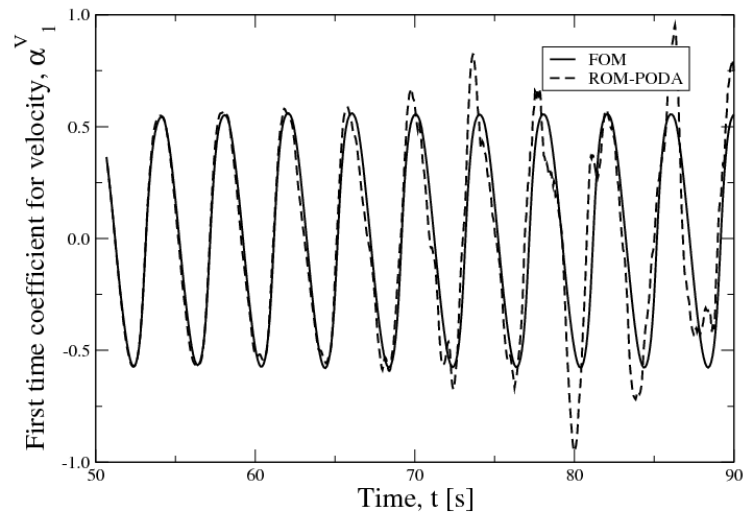
Figure 43: First time coefficient for velocity in the periodic region case.
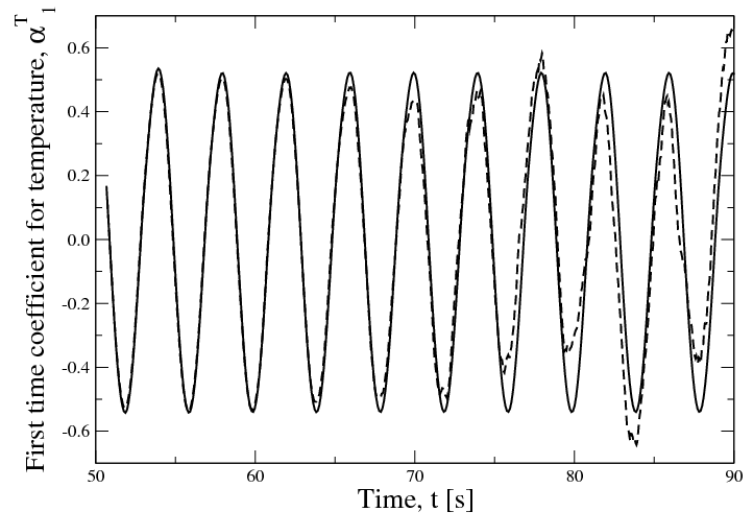
Figure 44: First time coefficient for temperature in the periodic region case.

CHAPTER VI

RANS SOLVER

The final goal of the current research is to implement a ROM for use with the in-house flow solver UNS3D. The first step in this implementation is a 2D inviscid ROM. The next section gives the physical model for fluid flow and notes the simplifications that will be made to obtain a 2D inviscid model. Next, the FOM, UNS3D, is discussed. Then the ROM that has been implemented is derived. Finally, verification results for the ROM are given and the current computational speed-up is discussed.

A. Physical Model

Fluid flow is governed by the conservation of mass, momentum and energy. In three-dimensions, for viscous flow, these conservation principles become the set of partial differential equations

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} + \frac{\partial \mathbf{G}}{\partial y} + \frac{\partial \mathbf{H}}{\partial z} = \mathbf{J} \tag{6.1}$$

where

$$U = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho E \end{pmatrix}, \qquad F = \begin{pmatrix} \rho u \\ \rho u^2 + p - \tau_{xx} \\ \rho u v - \tau_{xy} \\ \rho u w - \tau_{xz} \\ \rho E u + p u - k\frac{\partial T}{\partial x} - u\tau_{xx} - v\tau_{xy} - w\tau_{xz}, \end{pmatrix}$$

$$
G = \begin{pmatrix} \rho v \\ \rho u v - \tau_{yx} \\ \rho v^2 + p - \tau_{yy} \\ \rho v w - \tau_{yz} \\ \rho E v + p v - k\frac{\partial T}{\partial y} - u\tau_{yx} - v\tau_{yy} - w\tau_{yz}, \end{pmatrix}
$$

$$
H = \begin{pmatrix} \rho w \\ \rho u w - \tau_{zx} \\ \rho v w - \tau_{zy} \\ \rho w^2 + p - \tau_{zz} \\ \rho E w + p w - k\frac{\partial T}{\partial z} - u\tau_{zx} - v\tau_{zy} - w\tau_{zz}, \end{pmatrix}
$$

and

$$
J = \begin{pmatrix} 0 \\ \rho f_x \\ \rho f_y \\ \rho f_z \\ \rho(u f_x + v f_y + w f_z) + \rho\dot{q}. \end{pmatrix}.
$$

In practice, an inviscid test case will be used, meaning that the shear stress terms, τ in the above system of equations, will be neglected. In addition no source terms or body forces will be present so $\mathbf{J} = 0$

Since $\mathbf{U}$ is conservative, using the conservative variables for the POD basis functions will allow the time derivative to be simplified using the orthogonality of the basis functions. As a result, $\rho, \rho u, \rho v, \rho w$, and ρE were captured as snapshots and basis functions were computed.

B.   Full-Order Model: UNS3D

The integral form of the Navier-Stokes written in conservative variables is

$$\frac{\partial}{\partial t}\int_{\Omega}\mathbf{U}d\Omega + \oint_{S}\mathbf{F}\cdot\mathbf{n}dS = 0. \tag{6.2}$$

Here the control volume, Ω, is assumed to be stationary (non-rotational) and no source terms are present.

UNS3D solves a finite-volume discretization of the integral form of the conservative Navier-Stokes equations. This is expressed as

$$\frac{\Omega_i}{\Delta t_i}\Delta q_i = \sum_{j=1}^{\text{nedges}(i)} F_{ij}S_{ij} = \text{Res}_i \tag{6.3}$$

Past work has used UNS3D to simulate flow through turbomachinery [73] and for aeroelastic simulations [74]. UNS3D has also been used to solve acoustic problems in cavity flow [75]. Currently it is being adapted to allow for parallel computations and to simulate flow around wind turbines.

C.   Reduced-Order Model: UNS3DROM

If the discretization is constant in time, *i.e.*, the mesh is stationary, (6.3) can be rearranged and becomes

$$\Delta q_i = \frac{\Delta t_i}{\Omega_i}\sum_{j=1}^{\text{nedges}(i)} F_{ij}S_{ij} = \widetilde{\text{Res}}_i. \tag{6.4}$$

This is a very convenient form for reduction. For this problem the POD approximation is

$$q(x_i, t_k) = \varphi_0^q(x_i) + \sum_{j=1}^{m^q} \varphi_j^q(x_i)\alpha_j^q(t_k). \tag{6.5}$$

Substituting (6.5) into (6.4) gives the system,

$$\sum_{j=1}^{m^q} \varphi_j^q(x_i)\Delta\alpha_j^q = \frac{\Delta t_i}{\Omega_i} \sum_{j=1}^{nedges(i)} F_{ij}S_{ij} = \widetilde{\mathrm{Res}}_i. \tag{6.6}$$

This system is then projected onto the collection of POD basis functions, $\{\phi^q\}$, resulting the system of ordinary differential equations

$$\Delta\alpha_\ell^q = \left(\widetilde{\mathrm{Res}}_i, \phi_\ell^q \right) \tag{6.7}$$

where the orthogonality of the POD modes is used to simplify the left hand side.

D. Test Case: Channel with a Bump

To validate the ROM, a simple test case was selected consisting of a bump in a two-dimensional channel. The flow was subsonic and the channel was discretized using an inviscid mesh of 8100 nodes. The geometry and mesh for this test case are shown in Fig. 45. The flow parameters used for both the FOM and the ROM are given in the table on page 94. The FOM was used to simulate 10,000 time steps which corresponded to 0.13 s of physical time. During this interval, 100 snapshots were collected and used to compute the POD basis functions. In the ROM, 10 modes were used to model each field variable.

E. Results

Figures 46-49 show comparisons between the time coefficients obtained by projecting the snapshots of the FOM onto the basis functions to the time coefficients computed by integrating the ROM. Figures 50-53 show the time average mode, φ_0, and the first and second POD modes for each of the four field variables.
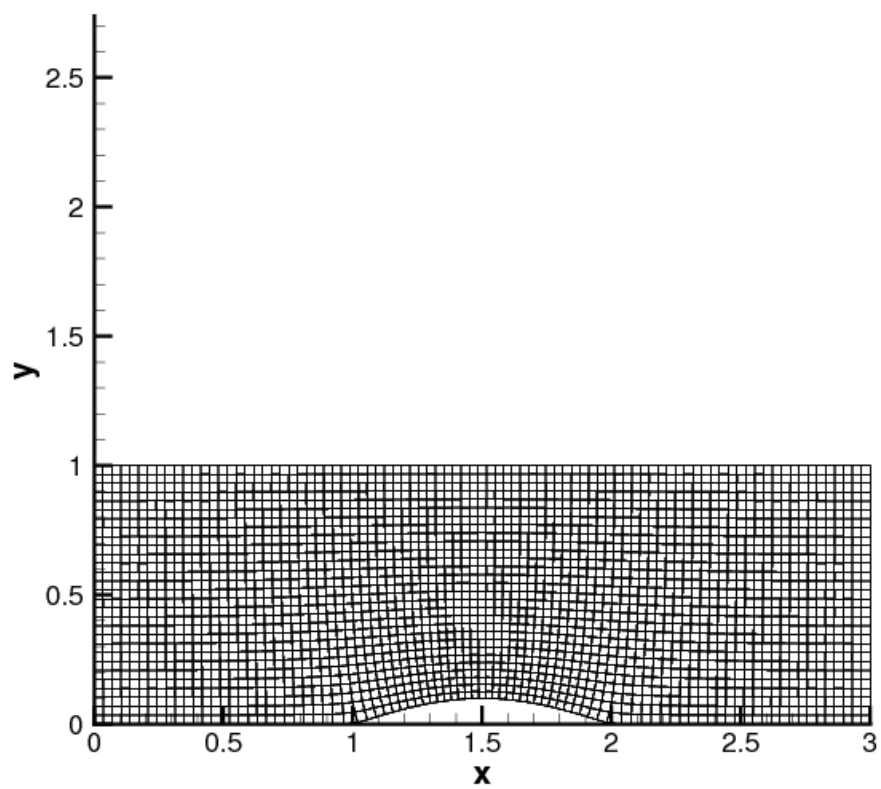
Figure 45: Inviscid mesh for channel with a bump.
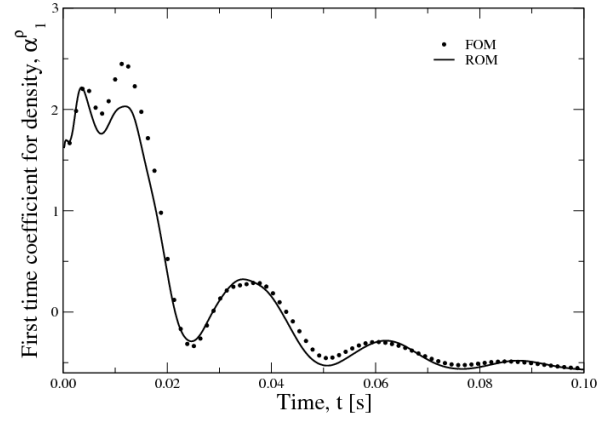
(a)
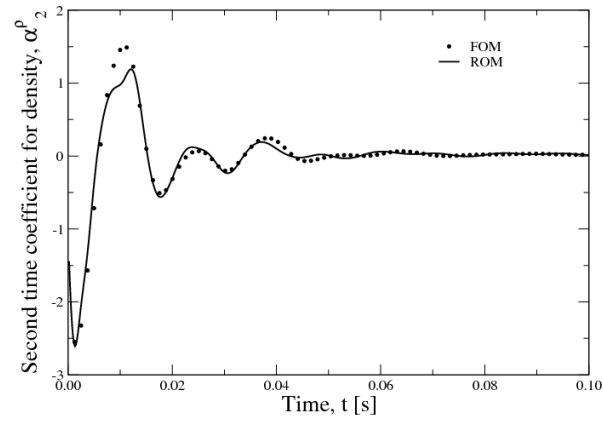
(b)

Figure 46: Comparison of time coefficients computed by projecting FOM snapshots directly on POD basis functions against time coefficient computed using the ROM for: (a) first density time coefficient and (b) second density time coefficient.
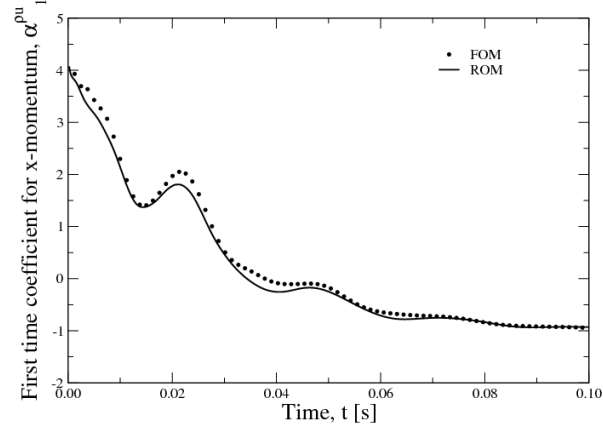
(a)
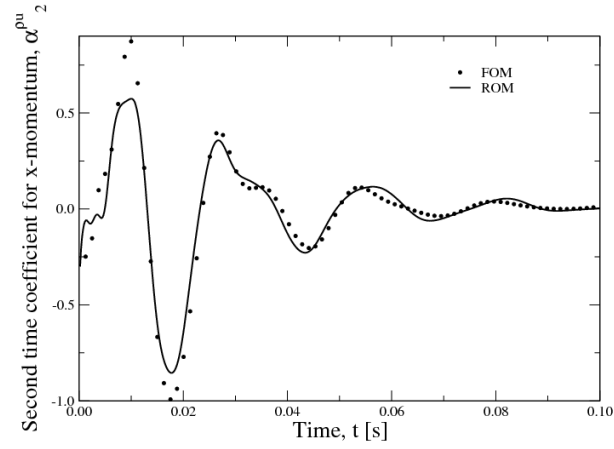
(b)

Figure 47: Comparison of time coefficients computed by projecting FOM snapshots directly on POD basis functions against time coefficient computed using the ROM for: (a) first x-direction momentum time coefficient and (b) second x-direction momentum time coefficient.
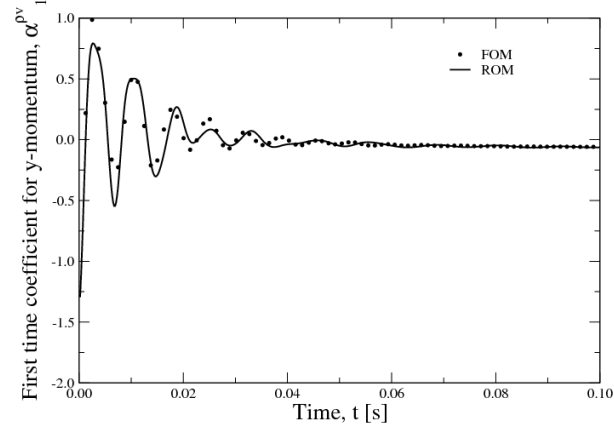
(a)
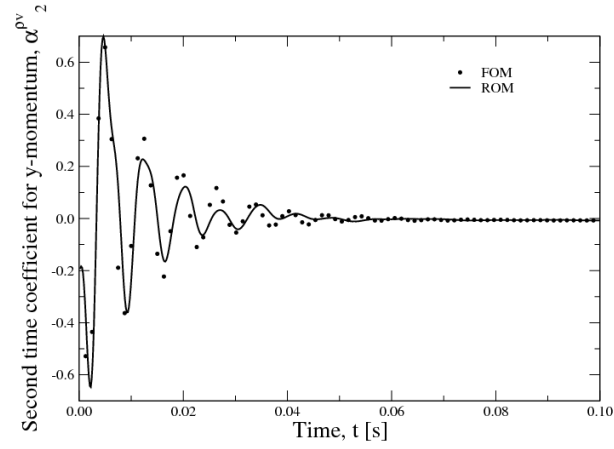
(b)

Figure 48: Comparison of time coefficients computed by projecting FOM snapshots directly on POD basis functions against time coefficient computed using the ROM for: (a) first y-direction momentum time coefficient and (b) second y-direction momentum time coefficient.
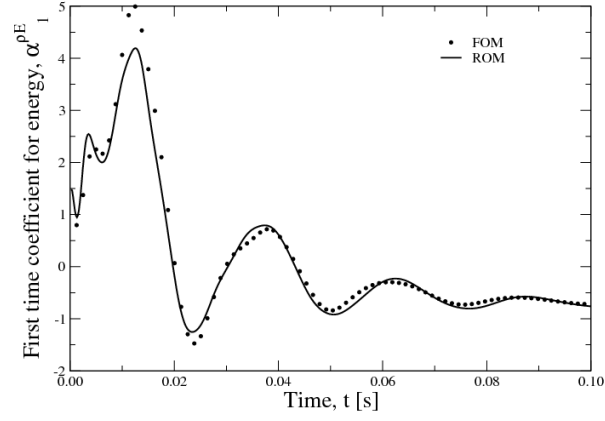
(a)
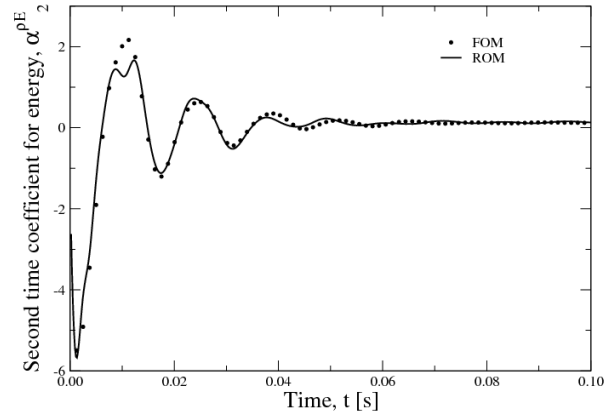
(b)

Figure 49: Comparison of time coefficients computed by projecting FOM snapshots directly on POD basis functions against time coefficient computed using the ROM for: (a) first energy time coefficient and (b) second energy time coefficient.
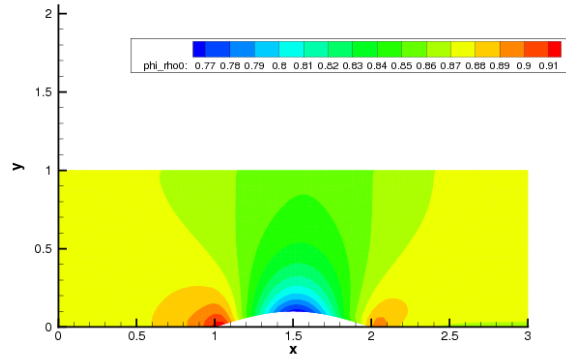
(a)
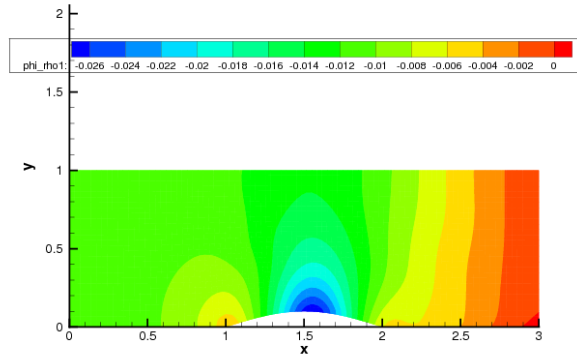
(b)
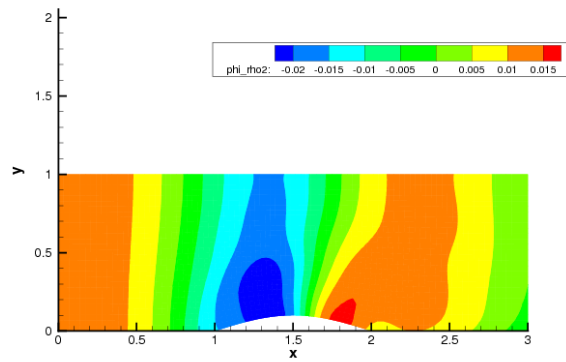
(c)

Figure 50: POD modes for density, ρ: (a) time average mode for density, ϕ_0^ρ, (b) first density POD mode, ϕ_1^ρ, and (c) second density POD mode, ϕ_2^ρ.

(a)

(b)

(c)
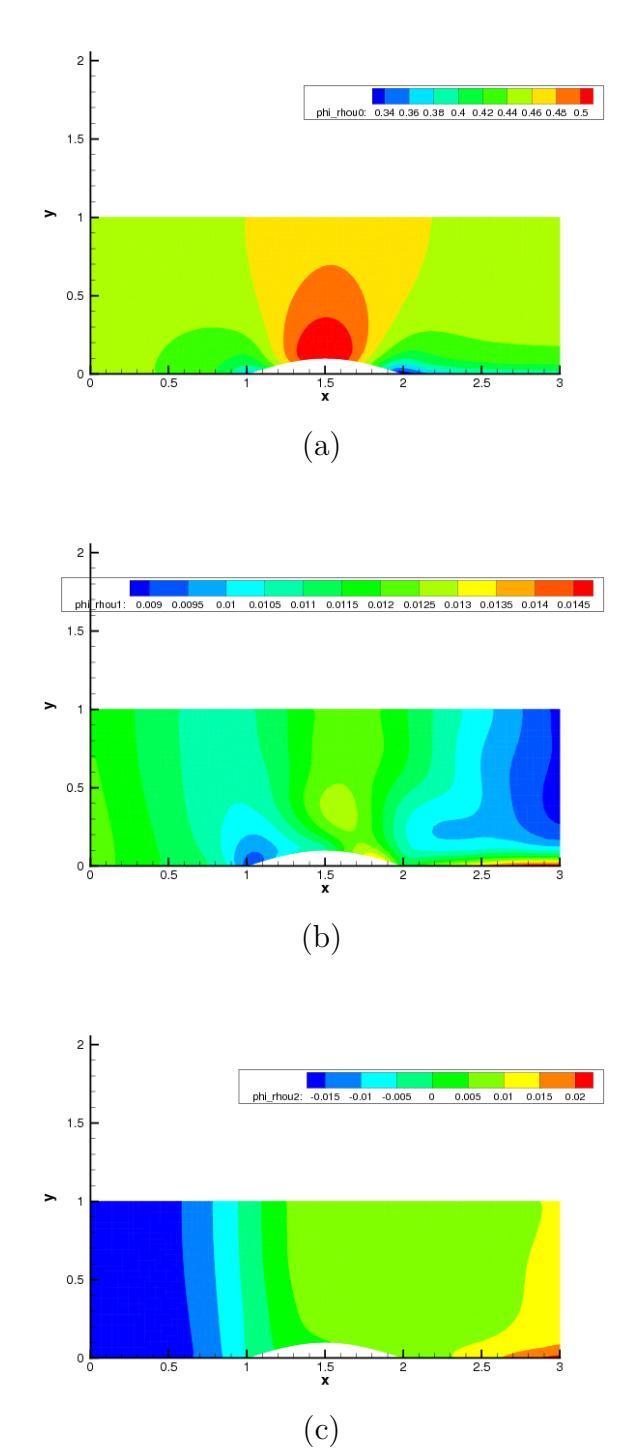
Figure 51: POD modes for x-direction momentum, ρu: (a) time average mode, $\phi_0^{\rho u}$, (b) first POD mode, $\phi_1^{\rho u}$, and (c) second POD mode, $\phi_2^{\rho u}$.
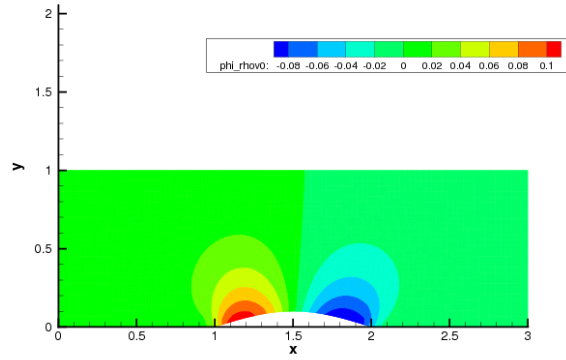
(a)
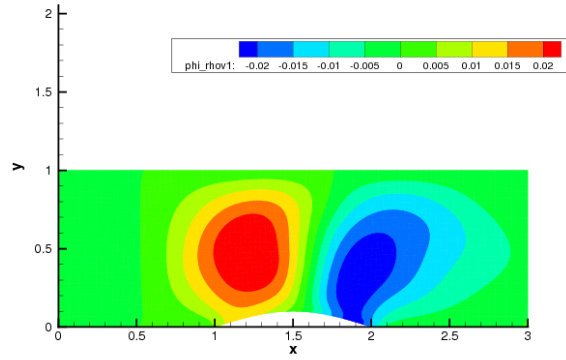
(b)
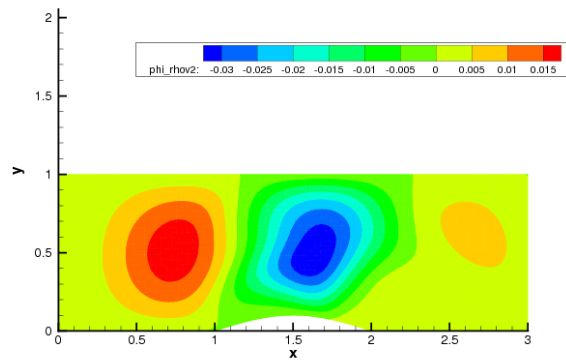
(c)

Figure 52: POD modes for y-direction momentum, ρv: (a) time average mode, $\phi_0^{\rho v}$, (b) first POD mode, $\phi_1^{\rho v}$, and (c) second POD mode, $\phi_2^{\rho v}$.
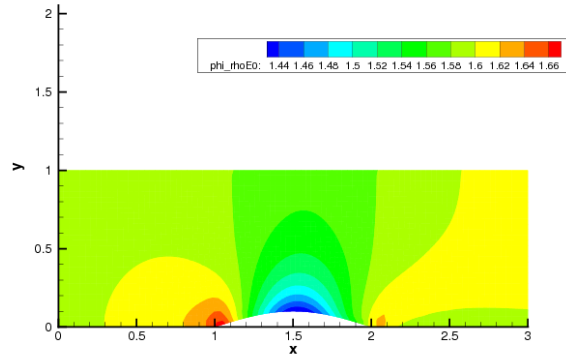
(a)
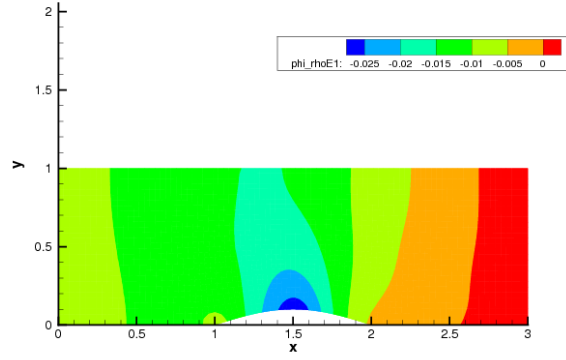
(b)
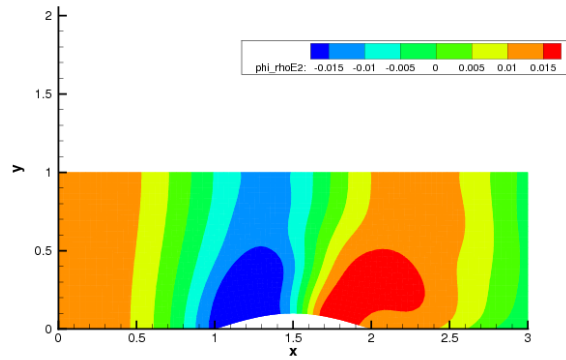
(c)

Figure 53: POD modes for energy, ρE: (a) time average mode, $\phi_0^{\rho E}$, (b) first POD mode, $\phi_1^{\rho E}$, and (c) second POD mode, $\phi_2^{\rho E}$.

Table VI: Parameters of a channel with a bump.

Parameter	Description	Units	
x_{length}	Length of the domain in x-direction	-	3.0 c
y_{length}	Length of the domain in y-direction	-	1.0 c
c	Length of bump	m	1.0
$nnode$	Number of nodes in grid	-	8100
$ncell$	Number of cells in grid	-	5162
p_{tot}	Total pressure	Pa	101325
p_{back}	Back pressure	Pa	79439.2
T_{tot}	Total temperature	K	289
u_0	Inlet Mach number	-	0.6

Figures 54-57 show the final solution for the four field variables computed by the FOM and ROM. Table VII summarizes the computational speed-up obtained for this ROM. The data used to compute the POD modes were generated with a CFL number of one. Through trial and error, it was found that the maximum CFL number allowable for this test case without divergence occurring was six. Also through trial and error, the ROM was found to converge with a maximum CFL number of fifteen. The computational speed-up factor of the ROM compared to the FOM with a CFL number of one was 8.1, while compared to the FOM with a CFL number of six was 1.35.

Table VII: Summary of the computational speed-up for UNS3DROM.

Method	FOM (1)	FOM (6)	ROM
CFL number	1	6	15
Total physical time simulated	0.13 s	0.76 s	0.95 s
CPU time	992 s	999 s	922 s
CPU/Physical	7826	1309	966

These results show that further development of a ROM using a POD-D approach for UNS3D is worthwhile. The current formulation relies entirely on the larger time step for its computational speed-up. The next step is to use the orthogonality of the POD modes to simplify the computation of the projection of the residual on the right-hand side of the governing set of ODEs. This remains to be done and is discussed further in the final chapter.
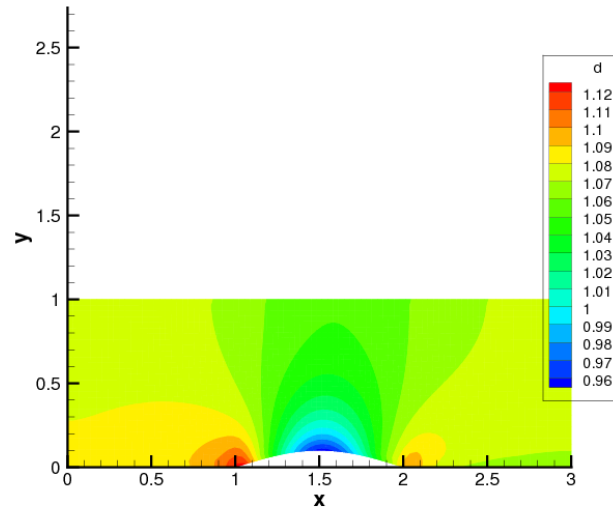
(a)
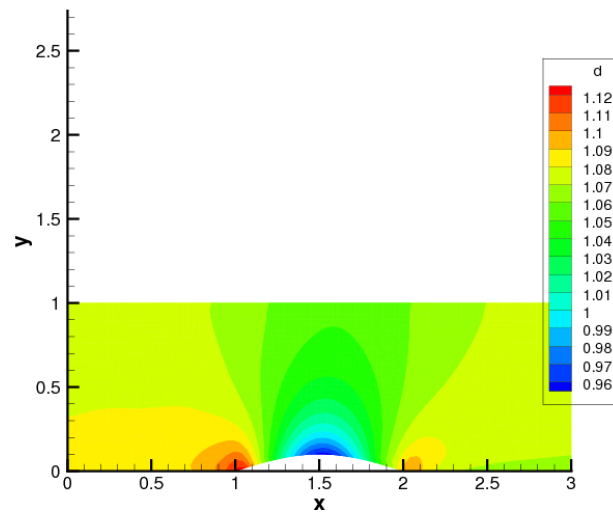
(b)

Figure 54: Contour plot of density at the steady state for: (a) FOM and (b) ROM.
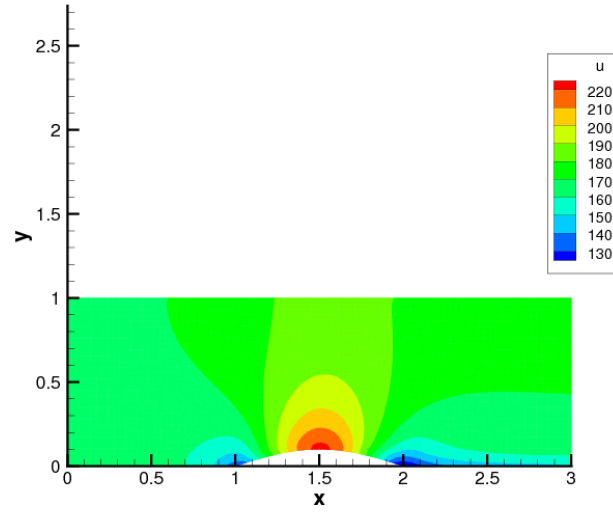
(a)
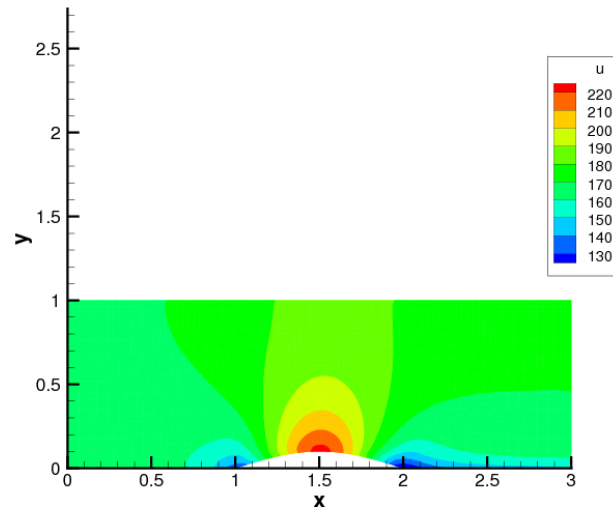
(b)

Figure 55: Contour plot of x-direction velocity at the steady state for: (a) FOM and (b) ROM.
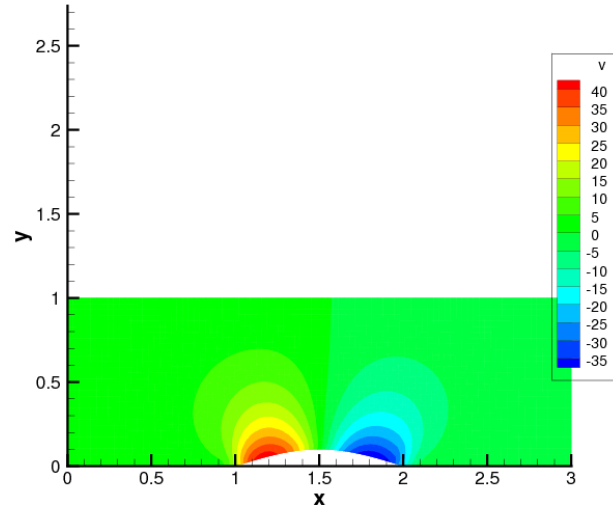
(a)
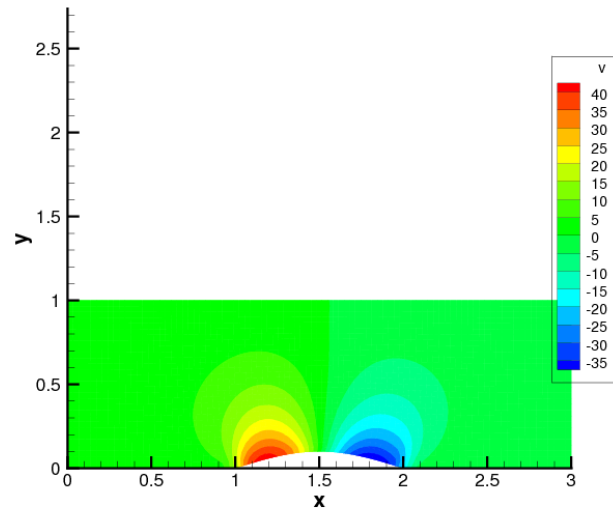
(b)

Figure 56: Contour plot of y-direction velocity at the steady state for: (a) FOM and (b) ROM.
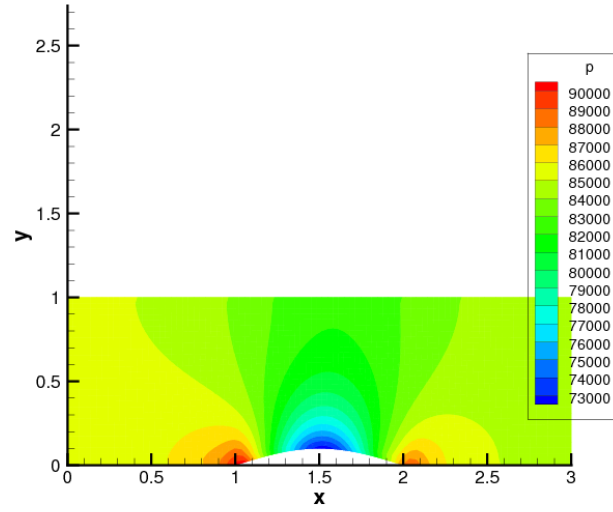
(a)
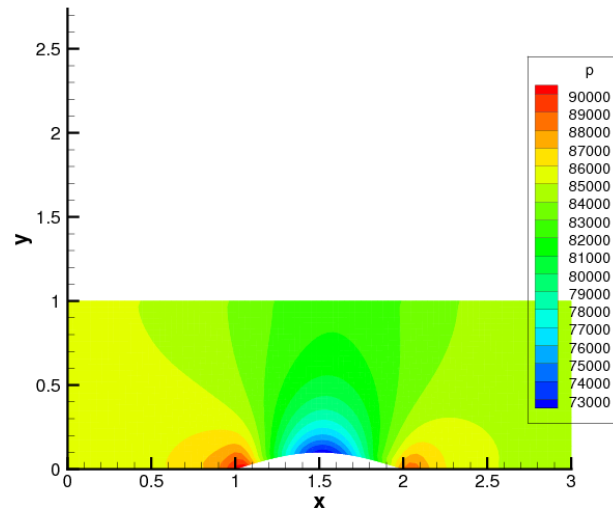
(b)

Figure 57: Contour plot of pressure at the steady state for: (a) FOM and (b) ROM.

CHAPTER VII

POD IMPLEMENTATION ASPECTS*

While implementing ROMs for various flow solvers, a number of practical aspects were addressed. Of particular interest were the best spacing of the snapshot used to compute the POD basis functions in time and the degree of coupling between field variables to consider when computing the POD basis functions. The next section presents a study used to evaluate several time sampling strategies for snapshots. Results are presented comparing the POD approximation error of the various approaches. After that, the method used to compute the autocorrelation matrix whose eigenvectors are the POD basis functions is discussed. A method for coupling the field variables when computing the autocorrelation matrix is compared to a split approach that considers each variable individually. The error in the POD approximation is computed and the approaches are compared.

A.   Snapshot Distribution

The literature makes no claim as to the optimal distribution of snapshots, but the important dynamics of the flow must be contained in the snapshot ensemble if they are to be captured by the basis functions. Examining (2.2), the least-square truncation error, ε_m, is a function of t_i. The task then becomes finding a set of values for t_i that minimizes ε_m for a given number of snapshots. Park and Lee [9] first proposed collecting snapshots more frequently during transient periods, but no attempt was made to quantify the error in this method or to compare it to an evenly spaced

---

*Part of this chapter is reprinted with permission from "Practical Aspects of the Implementation of Proper Orthogonal Decomposition" by T.A. Brenner, P.G.A. Cizmas, T.J. O'Brien and R.W. Breault, 47th AIAA Aerospace Sciences Meeting, Orlando, FL, 2009 by the authors.

ensemble.

Two studies were carried out herein to assess the effect of time sampling of snapshots on POD error. In the first, several snapshot databases were generated by collecting snapshots at a constant rate. The time between snapshots ranged from 0.1 s to 0.001 s, depending on the set. In total, ten sets were collected. These sets are listed in the table on page 106.

POD was performed on each snapshot database generated and the reconstructed solution was compared with a reference solution, $u(\mathbf{x}, t_i)$, consisting of a snapshot database containing 4823 snapshots. This reference solution represented the output of the FOM at every time step. It is therefore reasonable to assume that the 4823 snapshot database captures all of the relevant flow dynamics. The error for each snapshot database is then defined by (2.2) and the number of modes used in the reconstruction for each variable is given by Table VIII. For this snapshot distribution study, the $\alpha_k(t_i)$ terms in (2.2) were calculated using a projection method, given by

$$\alpha_k(t_i) = (u(\mathbf{x}, t_i), \varphi_k(\mathbf{x})), \qquad i = 1, \ldots, M. \tag{7.1}$$

In the second study, the snapshots were redistributed so that the snapshot frequency was increased during transient periods, while keeping the total number of snapshots approximately constant. The solution computed from the reconstruction of these data was then compared with the reference solution, again using (2.2). The increase in snapshot frequency during transience was accomplished in two ways. In the first case, two sampling rates were used: a larger constant sampling rate in the transient period and a smaller constant sampling rate in the quasi-steady period. Fig. 58 shows the time between snapshots as a function of the physical time and refers to this case as the step method. Park and Lee [9] have implemented a step method with five steps but given the two distinct regions present in the test case used

Table VIII: Number of modes used for isothermal flow and their symbols.

Field variable	Symbol	No. of modes
Gas pressure	m^{p_g}	2
Void fraction	m^{ϵ_g}	7
u gas velocity	m^{u_g}	2
v gas velocity	m^{v_g}	5
u solids velocity	m^{u_s}	8
v solids velocity	m^{v_s}	6

herein [14] two steps were deemed sufficient.

In the second case, referred to as the continuous method, a logarithmic function was used to give a continuously varying sampling rate, with snapshots collected less frequently as the simulation progressed.

To obtain this logarithmic distribution, the FOM was run to generate an evenly spaced snapshot database with 320 realizations. The flow was then simulated using the ROM and the relationship between physical time, t, and CPU time, t_{cpu} was captured. This relationship is shown in Fig. 59 and is assumed to be a measure of the time scale of the transport phenomena. Therefore, these data were selected as a useful measure of the complexity of the flow and a logarithmic function was found to approximate CPU time as a function of physical time. The function was expressed by

$$t_{cpu}(t) = 588.14 \ln(t) + 1071.0.$$

This equation was then divided by a constant factor to produce a sampling method that gathered approximately 200 snapshots. The sampling times were given
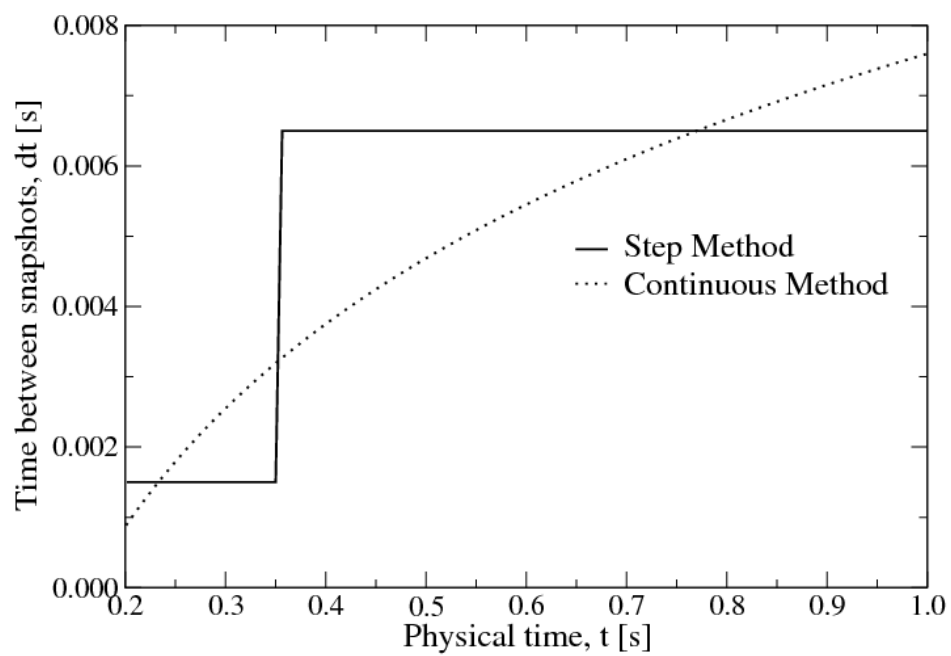
Figure 58: Time between snapshots as a function of physical time, t, for method 1: constant dt on each interval, and method 2: continuously varying dt.
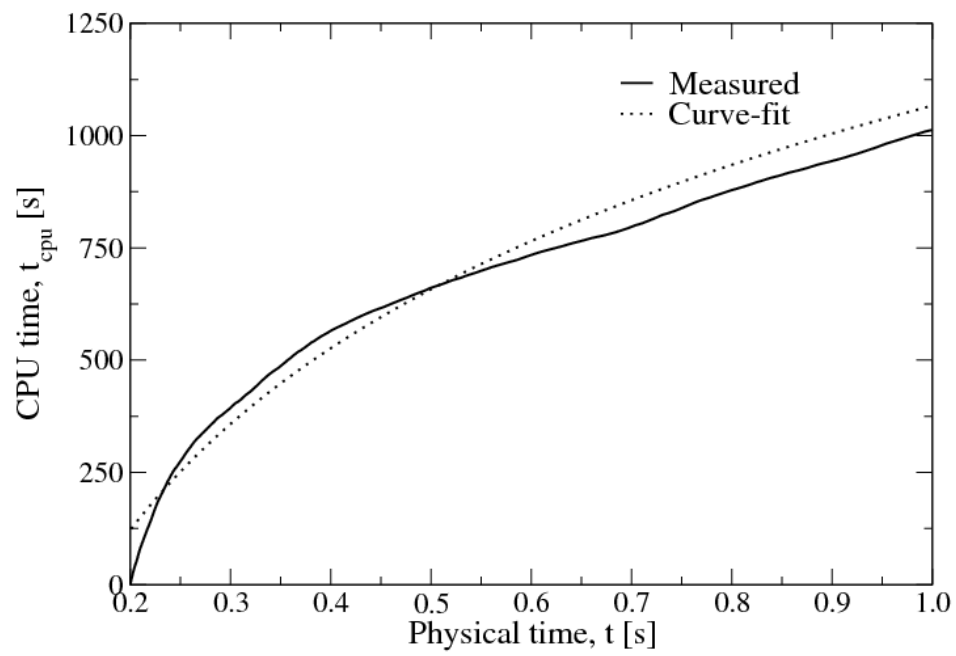
Figure 59: Relationship between physical time and CPU time in ROM: (a) as measured and (b) as a logarithmic function fit to the data.

iteratively as

$$t_{i+1} = t_i + dt(t_i),$$

where

$$dt(t_i) = 0.00419 \ln(t_i) + 0.00763.$$

Snapshot collection started at $t = 0.2$ s and continued until $t = 1.0$ s, resulting in the collection of 197 snapshots. Fig. 58 shows the time sampling of the snapshots given by this iterative process, where it is referred to as the continuous method. Once the snapshots were collected, POD was performed and the error was computed for each field variable using (2.2).

The errors computed using (2.2) for the snapshot distribution study are given in Table IX. To obtain these error values, the solution was reconstructed using the number of modes given in Table VIII. The reconstructed solutions were linearly interpolated in time to find values of the field variables at the proper time for comparison with the reference solution at all 4283 time locations.

As Table IX shows, the error tended to decrease as additional snapshots were collected and added to the computation of the basis functions. Note that the case with 200 snapshots has been presented out of sequence for easier comparison with the two unevenly distributed cases.

In the unevenly distributed cases, where the snapshots were collected more frequently in the areas of flow transience, the errors improved drastically as compared to snapshots that were evenly spaced with a similar number of snapshots collected. In the study, the best results for gas pressure were obtained using a logarithmic distribution of 197 snapshots. These results were generally better than even the case where 800 evenly distributed snapshots were used.

Using a distribution that was constant in the transient region and another, less

Table IX: POD reconstruction error for various snapshot distributions.

Time between snapshots, dt [s]	No. of snapshots	Error, ε					
		u_g	v_g	u_s	v_s	ϵ_g	p_g
0.1	8	0.166	48.628	4.324E-003	0.584	5.737E-006	3.424E+006
0.075	12	0.167	30.761	5.840E-003	0.518	5.305E-006	1.977E+006
0.05	16	0.166	58.420	6.459E-003	0.558	3.204E-006	3.978E+006
0.025	32	0.163	21.339	6.328E-003	0.478	6.342E-006	1.240E+006
0.01	80	0.161	14.513	1.539E-003	0.195	4.507E-007	6.779E+005
0.0075	108	0.160	1.165	5.346E-004	1.149E-002	1.438E-008	3.883E+004
0.005	160	0.161	1.709	1.052E-003	1.401E-002	1.578E-008	9.977E+004
0.0025	320	0.160	0.812	3.675E-004	2.059E-003	2.014E-009	9.807E+003
0.001	800	0.160	0.247	1.315E-004	1.937E-004	1.781E-009	1.287E+003
0.004	200	0.161	0.981	1.533E-003	9.015E-003	4.617E-009	4.406E+004
0.0015 and 0.0065	202	0.165	0.261	1.968E-004	2.488E-004	1.911E-008	1.456E+003
varied from 9.2E-4 to 0.0075	197	0.165	0.241	1.439E-004	3.525E-004	3.372E-008	1.254E+003

frequent, constant distribution in the quasi-steady region produced results that were nearly as good as the logarithmic distribution. This second approach is easier in practice, as the function that gave the logarithmic distribution in time required a curve fit to data calculated with the ROM and was therefore not known *a priori*. Thus, the method using different constant sampling rates at different times is recommended.

B. Composition of the Autocorrelation Matrix

At least two approaches exist for computing the autocorrelation matrix. A study was performed to evaluate two approaches for our test case. The two approaches explored herein were a coupled approach and a split approach.

Several coupled approaches have been developed. Kirby et al. [76] have proposed a simple coupling method by concatenating the field variables into a single vector-valued state variable. Lumley and Poje [77] derived a scaling factor for each variable to couple terms with different dimensions in compressible flows. This approach has been applied to fluid simulation [78, 79, 80, 81, 82, 53] and the analysis of experimental results [83] and is a more sophisticated coupled approach.

In the split approach, POD is performed on each field variable separately. No attempt is made to capture the coupling between the field variables when computing the POD modes. Unfortunately, no data have been presented comparing the POD reconstruction error for a coupled approach and a split approach.

For our coupled approach, we have chosen to concatenate the field variables into a single variable that was used to compute a single autocorrelation matrix. To avoid the large differences between the magnitude of various components of the state vector, $\mathbf{u}$, the data have been normalized by the maximum value of each component. That

is

$$\tilde{\aleph} = \frac{\aleph}{\aleph_{\max}}$$

where $\aleph$ represents any component of the state vector,

$$\tilde{\mathbf{u}}(\mathbf{x}, t_i) = \begin{pmatrix} \tilde{u}_g(\mathbf{x}, t_i) \\ \tilde{v}_g(\mathbf{x}, t_i) \\ \tilde{u}_s(\mathbf{x}, t_i) \\ \tilde{v}_s(\mathbf{x}, t_i) \\ \tilde{\epsilon}_g(\mathbf{x}, t_i) \\ \tilde{p}_g(\mathbf{x}, t_i) \end{pmatrix}.$$

This normalization step was necessary to provide an autocorrelation matrix with a more favorable condition number. In practice, without normalization, the pressure data are six orders of magnitude larger than the void fraction data, leading to an eigenvalue problem that is impossible to solve accurately.

The concatenated variable was then substituted into (2.4) and the coupled autocorrelation matrix, $\overline{\overline{R}}_c \in \mathbb{R}^{6n} \times \mathbb{R}^{6n}$, was given by

$$\overline{\overline{R}}_c = \begin{pmatrix} \overline{\overline{R}}_{11} & \overline{\overline{R}}_{12} & \dots & \overline{\overline{R}}_{1N} \\ \overline{\overline{R}}_{21} & \overline{\overline{R}}_{22} & \dots & \overline{\overline{R}}_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ \overline{\overline{R}}_{N1} & \overline{\overline{R}}_{N2} & \dots & \overline{\overline{R}}_{NN} \end{pmatrix}$$

where the submatrix $\bar{\bar{R}}_{ij} \in \mathbb{R}^6 \times \mathbb{R}^6$ was given by

$$
\bar{\bar{R}}_{ij} =
\begin{pmatrix}
\langle \tilde{p}_{gi} * \tilde{p}_{gj} \rangle & \langle \tilde{\epsilon}_{gi} * \tilde{p}_{gj} \rangle & \langle \tilde{u}_{gi} * \tilde{p}_{gj} \rangle & \langle \tilde{v}_{gi} * \tilde{p}_{gj} \rangle & \langle \tilde{u}_{si} * \tilde{p}_{gj} \rangle & \langle \tilde{v}_{si} * \tilde{p}_{gj} \rangle \\
\langle \tilde{p}_{gi} * \tilde{\epsilon}_{gj} \rangle & \langle \tilde{\epsilon}_{gi} * \tilde{\epsilon}_{gj} \rangle & \langle \tilde{u}_{gi} * \tilde{\epsilon}_{gj} \rangle & \langle \tilde{v}_{gi} * \tilde{\epsilon}_{gj} \rangle & \langle \tilde{u}_{si} * \tilde{\epsilon}_{gj} \rangle & \langle \tilde{v}_{si} * \tilde{\epsilon}_{gj} \rangle \\
\langle \tilde{p}_{gi} * \tilde{u}_{gj} \rangle & \langle \tilde{\epsilon}_{gi} * \tilde{u}_{gj} \rangle & \langle \tilde{u}_{gi} * \tilde{u}_{gj} \rangle & \langle \tilde{v}_{gi} * \tilde{u}_{gj} \rangle & \langle \tilde{u}_{si} * \tilde{u}_{gj} \rangle & \langle \tilde{v}_{si} * \tilde{u}_{gj} \rangle \\
\langle \tilde{p}_{gi} * \tilde{v}_{gj} \rangle & \langle \tilde{\epsilon}_{gi} * \tilde{v}_{gj} \rangle & \langle \tilde{u}_{gi} * \tilde{v}_{gj} \rangle & \langle \tilde{v}_{gi} * \tilde{v}_{gj} \rangle & \langle \tilde{u}_{si} * \tilde{v}_{gj} \rangle & \langle \tilde{v}_{si} * \tilde{v}_{gj} \rangle \\
\langle \tilde{p}_{gi} * \tilde{u}_{sj} \rangle & \langle \tilde{\epsilon}_{gi} * \tilde{u}_{sj} \rangle & \langle \tilde{u}_{gi} * \tilde{u}_{sj} \rangle & \langle \tilde{v}_{gi} * \tilde{u}_{sj} \rangle & \langle \tilde{u}_{si} * \tilde{u}_{sj} \rangle & \langle \tilde{v}_{si} * \tilde{u}_{sj} \rangle \\
\langle \tilde{p}_{gi} * \tilde{v}_{sj} \rangle & \langle \tilde{\epsilon}_{gi} * \tilde{v}_{sj} \rangle & \langle \tilde{u}_{gi} * \tilde{v}_{sj} \rangle & \langle \tilde{v}_{gi} * \tilde{v}_{sj} \rangle & \langle \tilde{u}_{si} * \tilde{v}_{sj} \rangle & \langle \tilde{v}_{si} * \tilde{v}_{sj} \rangle
\end{pmatrix}
$$

with $i, j = 1, \ldots, N$. The notation $\langle \aleph_i * \aleph_j \rangle$ is defined as

$$
\langle \aleph_i * \aleph_j \rangle = \frac{1}{M} \sum_{k=1}^{M} \aleph(x_i, t_k) \aleph(x_j, t_k)
$$

where $\aleph$ is any field variable, N is the number of nodes and M is the number of snapshots.

In practice, the method of snapshots was used to compute the POD basis functions. To obtain an equivalent coupling, the inner product used to compute the autocorrelation matrix C in the method of snapshots is

$$
(\tilde{\mathbf{u}}(\mathbf{x}, t_i), \tilde{\mathbf{u}}(\mathbf{x}, t_i)) = \int_D \Big(\tilde{u}_g(\mathbf{x}, t_i) \tilde{u}_g(\mathbf{x}, t_j) + \tilde{v}_g(\mathbf{x}, t_i) \tilde{v}_g(\mathbf{x}, t_j) + \tilde{u}_s(\mathbf{x}, t_i) \tilde{u}_s(\mathbf{x}, t_j)
$$

$$
+ \tilde{v}_s(\mathbf{x}, t_i) \tilde{v}_s(\mathbf{x}, t_j) + \tilde{\epsilon}_g(\mathbf{x}, t_i) \tilde{\epsilon}_g(\mathbf{x}, t_j) + \tilde{p}_g(\mathbf{x}, t_i) \tilde{p}_g(\mathbf{x}, t_j) \Big) \mathrm{d}\mathbf{x}. \quad (7.2)
$$

In the split approach, a separate C was computed for each field variable. This is analogous to computing an autocorrelation matrix with the off-diagonal sub-matrices

set to zero, or

$$
\overline{\overline{R}}_s = \begin{pmatrix}
\overline{\overline{R}}_{\langle p_g * p_g \rangle} & 0 & 0 & 0 & 0 & 0 \\
0 & \overline{\overline{R}}_{\langle \epsilon_g * \epsilon_g \rangle} & 0 & 0 & 0 & 0 \\
0 & 0 & \overline{\overline{R}}_{\langle u_g * u_g \rangle} & 0 & 0 & 0 \\
0 & 0 & 0 & \overline{\overline{R}}_{\langle v_g * v_g \rangle} & 0 & 0 \\
0 & 0 & 0 & 0 & \overline{\overline{R}}_{\langle u_s * u_s \rangle} & 0 \\
0 & 0 & 0 & 0 & 0 & \overline{\overline{R}}_{\langle v_s * v_s \rangle}
\end{pmatrix}
$$

where

$$
\overline{\overline{R}}_{\langle \aleph * \aleph \rangle} = \frac{\sum_{i=1}^{M} \aleph(\mathbf{x}, t_i) \aleph^T(\mathbf{y}, t_i)}{M}
$$

and, $\overline{\overline{R}}_s \in \mathbb{R}^{6n} \times \mathbb{R}^{6n}$.

A set of basis functions was calculated for each of the two approaches. The solutions were then reconstructed using (2.1), where the time coefficients were calculated using the projection method given by (7.1). The errors for each approach, as given by (2.2), were then compared to determine the better method. The results are presented in Table X. Also given is a relative error, defined as $\varepsilon_{rel} = \varepsilon / \aleph_{max}$.

As Table X shows, the POD reconstruction error was larger for the coupled approach than for the split approach for each variable—sometimes significantly so. For the gas and solids velocities the difference in error between the two methods is only about an order of magnitude. Both the gas pressure and void fraction, however, show a five order of magnitude difference in their error.

In the case of the void fraction this difference is not very significant. The void fraction can be expressed as

$$
\epsilon_g(\mathbf{x}, t_i) = \varphi_0^{\epsilon_g}(\mathbf{x}) + \sum_{k=1}^{M} \alpha_k^{\epsilon_g}(t_i) \varphi_k^{\epsilon_g}(\mathbf{x}), \qquad i = 1, \dots, M.
$$

The $\varphi_0^{\epsilon_g}(\mathbf{x})$ term is the time-average of the solution and is identical for both ap-

Table X: POD reconstruction error for coupled and split approaches.

Variable	Error			
	$\varepsilon^{coupled}$	ε^{split}	$\varepsilon^{coupled}_{rel}$	$\varepsilon^{split}_{rel}$
p_g	1.245E+006	57.91	1.230	5.72E-005
ϵ_g	1.012E-005	5.891E-010	1.012E-005	5.891E-010
u_g	1.305E-002	5.081E-003	9.758E-004	3.799E-004
v_g	1.380	8.980E-003	8.944E-002	5.820E-004
u_s	4.251E-005	4.833E-006	3.271E-004	3.719E-005
v_s	2.248E-004	4.866E-005	1.613E-003	3.491E-004

proaches. This time-average is always positive and less than one. The error calculated for the coupled approach was five orders of magnitude smaller than this average and the error for the split approach was ten orders of magnitude less. In both approaches the error is small enough to be considered insignificant.

For the gas pressure, however, the difference in error is significant. The error in the coupled approach is on the same order as $\varphi_0^{p_g}$ ($\sim 10^6$) while the error in the split approach is only on the order of 10. The difference in error between the two approaches is therefore quite significant. Clearly the split approach is superior for this test case.

CHAPTER VIII

CONCLUSIONS AND FUTURE WORK

This chapter presents the conclusions drawn from the results given in the preceding chapters. The next steps needed to continue this work are summarized. Topics for future study are discussed.

A. Conclusions

Despite advances in both computing hardware and computational algorithms, the simulation of certain fluid flows remains too complex for use in design and control. Ongoing research into ROMs shows promise for providing high-fidelity solutions at a fraction of the computational cost of the FOM. Many ROMs use spatial modes to transform PDEs into ODEs. Proper orthogonal decomposition has been shown to provide the optimal basis functions for these ROMs.

The research presented is a combination of the application of ROMs to novel problems and solutions to practical aspects arising in the implementation of these ROMs.

Previous work has constructed a ROM for two-dimensional isothermal multiphase flow. This ROM has been extended to include three-dimensional flows and non-isothermal flows. Validation results are presented for a minimum fluidization test case. These results show good agreement with the FOM at reference conditions.

While constructing the ROM for multiphase flow, difficulties arose in the simulation of bubbling flow. Bubbles are modeled as moving discontinuities in the void fraction. These moving discontinuities led to unphysical approximations for the void fraction in the ROM.

A method for augmenting the POD bases with point modes was proposed. In this

method, modes with value one at a single point were added to the collection of POD basis functions, which were masked at this point, to better capture discontinuities. Results were shown for a simple test case where point modes were used for the gas velocity, showing that the original solution was recovered.

A simple test problem with a single moving discontinuity was developed, using the transport, or one-dimensional wave, equation. This is the simplest model for which moving discontinuities are possible. Both a FOM and ROM were constructed for this problem. The ROM contained a shock mode which contained a discontinuity that was allowed to move in time to match the location of the discontinuity in the FOM. This approach was shown to remove the unphysical oscillations usually contained in POD-based ROMs for problems with moving discontinuities.

The difference between two approaches to deriving POD-based ROMs, POD-A and POD-D, were discussed and the approaches were compared. In the POD-A approach, the POD approximations are substituted into the governing PDEs prior to discretization. The differential terms are then transferred directly to the modes and time coefficients, prior to the Galerkin projection. This gives added flexibility to the ROM and allows greater freedom when choosing the integration scheme and the method for approximating spatial derivatives. In the POD-D approach, the POD approximations are substituted into the discretized form of the governing PDEs and the resulting algebraic equations are projected onto the POD basis functions using a Galerkin projection. This results in the ROM mimicking the integration strategy and approximation for the spatial derivatives used by the FOM. It seems in practice that this approach produces ROMs that are more likely to be stable than the POD-A approach.

A quasi-2D subsonic nozzle flow problem was selected to test the practical differences between the POD-A and POD-D approaches. A FOM based on the Mac-

Cormack scheme was implemented for this problem. ROMs were then derived and constructed using both the POD-A and POD-D approaches. The stability of each model was tested and the POD-D method was judged to be both more straightforward to implement and less likely to suffer from stability problems, manifested in this case by a stiff system of equations in the POD-A approach.

A POD-D approach was used to construct a ROM for two-dimensional inviscid flow. This ROM was based on the in-house finite-volume solver UNS3D. A simple test case consisting of a channel with a bump was used to validate the ROM. Excellent agreement was shown with the FOM for this case. An overall decrease in the computational time required to solve this problem was shown using the ROM as compared to the FOM.

Finally, two practical aspects of the implementation of ROMs were discussed. The first was the best placement in time of the snapshots from which the POD basis functions were computed. Using a multiphase flow test case, a decrease in the POD approximation error was achieved by clustering the snapshots at times of greater flow complexity. The second practical aspect was the form of the autocorrelation matrix used to compute the POD basis functions. Two approaches were evaluated: split and coupled. In the split approach, basis functions were computed for the field variables individually. In the coupled problem, the cross-correlation terms between field variables were considered. For the multiphase flow case considered, the POD approximation error was shown to be less when using the split approach. This was somewhat unexpected and may have been due to the way in which the variables were coupled. The proper coupling of void fraction terms remains an topic for future research.

B.   Future Work

A number of topics remain for future exploration. Further development is needed for the point mode POD method and the general augmentation method for solving problems with moving discontinuities. The development of a ROM for UNS3D is ongoing and each new POD implementation brings new practical problems to be solved.

1.   Point Mode POD

Although point mode POD was shown to be effective for recovering the correction solution when applied to the gas velocity in multiphase flow, it remains to be implemented to the computation of the void fraction. The current algorithm solves for void fraction corrections, instead of void fraction itself. In order to implement point modes, a new algorithm is needed to solve for void fraction directly. This remains a topic for future study.

2.   Augmentation with Discontinuity Modes

The augmentation of the POD bases with discontinuity modes was shown to be effective for a simple one-dimensional problem. Further work is needed to generalize this process to two- and three-dimensions. Also, methods are needed to compose proper shock modes for cases where the location and shape of the discontinuities are unknown.

3.   ROM for UNS3D

The initial implementation of a ROM for UNS3D was for an inviscid, two-dimensional problem using an explicit solver. The next step is to extend the ROM to include

three-dimensional and viscous flow. After that, an implicit solver will need to be implemented. Also, the current implementation reconstructs the field variables, then computes the residual, then projects that residual onto the POD basis functions. This is very inefficient. A greater speed-up is achievable by pre-projecting the basis functions contained within the residual, taking advantage of the orthogonality of the modes, to build a subroutine that computes the $\Delta\alpha$ terms directly from the time coefficients, making the reconstruction of the field variables unnecessary.

4. Practical Aspects of POD

In the snapshot distribution study, an effort was made to identify a measure of complexity for the flow considered and base the snapshot sampling on this measure. This measure, however, was rather arbitrary. Further work is needed to identify the optimal placement of the snapshots in time, as opposed to the trial and error method developed here.

Further work is also needed to develop a dimensionally-correct method for coupling the void fraction to the other field variables in the computation of the POD basis functions when using a coupled approach for composing the autocorrelation matrix. A proper method for coupling the variables used in the POD-based ROM for UNS3D is also needed. Finally, this coupling method needs to be tested against a split approach for the UNS3D variables.

REFERENCES

[1] E. H. Dowell, K. C. Hall, J. P. Thomas, R. Florea, B. Epureanu, J. Heeg, Reduced order models in unsteady aerodynamics, in: 40th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference and Exhibit, St. Louis, MO, 1999.

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

[3] R. Florea, K. C. Hall, Reduced order modeling of unsteady flows about airfoils, in: Aeroelasticity and Fluid Structure Interaction Problems, ASME International Mechanical Engineering Congress and Exposition, 1994, 49–68.

[4] K. C. Hall, R. Florea, P. J. Lanzkron, A reduced order model of unsteady flows in turbomachinery, J. Turbomach. 117 (3) (1995) 375–383.

[5] K. C. Hall, Eigenanalysis of unsteady flows about airfoils, cascades, and wings, AIAA J. 32 (12) (1994) 2426–2432.

[6] R. Florea, K. C. Hall, P. G. A. Cizmas, Eigenmode analysis of unsteady viscous flows in turbomachinery cascades, in: T. H. Fransson (Ed.), Proceedings of the 8th International Symposium on Unsteady Aerodynamics and Aeroelasticity of Turbomachines, Stockholm, Sweden, 1997, 767–782.

[7] R. Florea, K. C. Hall, P. G. A. Cizmas, Reduced-order modeling of unsteady viscous flow in a compressor cascade, AIAA J. 36 (6) (1998) 1039–1048.

[8] A. Bangia, P. Batcho, I. Kevrekidis, G. Karniadakis, Unsteady two-dimensional

flows in complex geometries: Comparative bifurcation studies with global eigen-function expansions, SIAM J. Sci. Comput. 18 (3) (1997) 775–805.

[9] H. M. Park, M. W. Lee, An efficient method of solving the Navier-Stokes equations for flow control, Int. J. Numer. Meth. Eng. 41 (6) (1998) 1133–1151.

[10] P. G. A. Cizmas, A. Palacios, Proper orthogonal decomposition of turbine rotor-stator interaction, J. Propul. Power 19 (2) (2003) 268–281.

[11] Y. Utturkar, B. N. Zhang, W. Shyy, Reduced-order description of fluid flow with moving boundaries by proper orthogonal decomposition, Int. J. Heat Fluid Fl. 26 (2) (2005) 276–288.

[12] P. G. A. Cizmas, A. Palacios, T. O'Brien, M. Syamlal, Proper-orthogonal decomposition of spatio-temporal patterns in fluidized beds, Chem. Eng. Sci. 58 (19) (2003) 4417–4427.

[13] T. Yuan, P. G. Cizmas, T. O'Brien, A reduced-order model for a bubbling fluidized bed based on proper orthogonal decomposition, Comput. Chem. Eng. 30 (2) (2005) 243–259.

[14] P. G. A. Cizmas, B. R. Richardson, T. Brenner, T. J. O'Brien, R. W. Breault, Acceleration techniques for reduced-order models based on proper orthogonal decomposition, J. Comput. Phys. 227 (2008) 7791–7812.

[15] P. Ding, X. Wu, Y. He, W. Tao, A fast and efficient method for predicting fluid flow and heat transfer problems, J. Heat Transf. 130 (3) (2008) 1–1.

[16] C. Farhat, D. Amsallem, Recent advances in reduced-order modeling and application to nonlinear computational aeroelasticity, in: 46th AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, 2008.

[17] C. Homescu, L. R. Petzold, R. Serban, Error estimation for reduced-order models of dynamical systems, SIAM J. Numer. Anal. 43 (4) (2005) 1693–1714.

[18] A.-x. Wang, Y.-C. Ma, An error estimate of the proper orthogonal decomposition in model reduction and data compression, Numer. Meth. Part Differ. Equ. 25 (4) (2009) 972–989.

[19] Y. Chen, K. A. Hoo, Uncertainty propagation for effective reduced-order model generation, Comput. Chem. Eng. 34 (10) (2010) 1597–1605.

[20] D. Rempfer, On low-dimensional Galerkin models for fluid flow, Theor. Comput. Fluid Dyn. 14 (2) (2000) 75–88.

[21] M. Giles, Stability analysis of a Galerkin/Runge-Kutta Navier-Stokes discretisation on unstructured tetrahedral grids, J. Comput. Phys. 132 (2) (1997) 201–214.

[22] M. F. Barone, I. Kalashnikova, D. J. Segalman, H. K. Thornquist, Stable Galerkin reduced order models for linearized compressible flow, J. Comput. Phys. 228 (6) (2009) 1932–1946.

[23] A. Iollo, S. Lanteri, J. Desideri, Stability properties of POD-Galerkin approximations for the compressible Navier-Stokes equations, Theor. Comp. Fluid Dyn. 13 (2000) 377–396.

[24] V. L. Kalb, A. E. Deane, An intrinsic stabilization scheme for proper orthogonal decomposition based low-dimensional models, Phys. Fluids 19 (5) (2007) 1–18.

[25] X. Gloerfelt, Compressible proper orthogonal decomposition/Galerkin reduced-order model of self-sustained oscillations in a cavity, Phys. Fluid 20 (2008) 1–22.

[26] S. Sirisup, G. Karniadakis, A spectral viscosity method for correcting the long-term behavior of POD models, J. Comput. Phys. 194 (1) (2004) 92–116.

[27] B. Noack, K. Afanasiev, M. Morzynski, G. Tadmor, F. Thiele, A hierarchy of low-dimensional models for the transient and post-transient cylinder wake, J. Fluid Mech. 497 (2003) 335–363.

[28] E. H. Dowell, D. Tang, Dynamics of Very High Dimensional Systems, World Scientific Publ. Co., New Jersey, 2003.

[29] D. J. Lucia, P. S. Beran, W. A. Silva, Reduced-order modeling: New approaches for computational physics, Prog. Aerosp. Sci. 40 (1-2) (2004) 51–117.

[30] D. Lucia, M. Pachter, P. Beran, Rocket nozzle flow control using a reduced-order computational fluid dynamics model, J. Guid. Control Dynam. 25 (3) (2002) 449–454.

[31] B. Epureanu, E. Dowell, K. Hall, Reduced-order models of unsteady transonic viscous flows in turbomachinery, J. Fluid Struct. 14 (8) (2000) 1215–1234.

[32] J. Anttonen, P. King, P. Beran, POD-based reduced-order models with deforming grids, Math. Comput. Model. 38 (1-2) (2003) 41–62.

[33] J. Anttonen, P. King, P. Beran, Applications of multi-POD to a pitching and plunging airfoil, Math. Comput. Model. 42 (3-4) (2005) 245–259.

[34] G. Lewin, H. Haj-Hariri, Reduced-order modeling of a heaving airfoil, AIAA J. 43 (2) (2005) 270–283.

[35] R. Bourguet, M. Braza, A. Dervieux, Reduced-order modeling of transonic flows around an airfoil submitted to small deformations, J. Comput. Phys. 230 (1) (2011) 159 – 184.

[36] K. Nakakita, S. Nadarajah, W. Habashi, Toward real-time aero-ecing simulation of complete aircraft via FENSAP-ICE, J. Aircraft 47 (1) (2010) 96–109.

[37] B. J. O'Donnell, B. T. Helenbrook, Proper orthogonal decomposition and incompressible flow: An application to particle modeling, Comput. Fluids 36 (7) (2007) 1174–1186.

[38] A. Qamar, S. Sanghi, Steady supersonic flow-field predictions using proper orthogonal decomposition technique, Comput. Fluids 38 (6) (2009) 1218–1231.

[39] Z. Luo, Y. Zhou, X. Yang, A reduced finite element formulation based on proper orthogonal decomposition for Burgers equation, Appl. Numer. Math. 59 (8) (2009) 1933–1946.

[40] M. A. Singer, W. H. Green, Using adaptive proper orthogonal decomposition to solve the reaction-diffusion equation, Appl. Numer. Math. 59 (2) (2009) 272–279.

[41] R. Bourguet, M. Braza, A. Sevrain, A. Bouhadji, Capturing transition features around a wing by reduced-order modeling based on compressible Navier-Stokes equations, Phys. Fluids 21 (9) (2009) 094104.

[42] C. E. Tinney, M. N. Glauser, E. L. Eaton, J. A. Taylor, Low-dimensional azimuthal characteristics of suddenly expanding axisymmetrical flows, J. Fluid Mech. 567 (2006) 141–155.

[43] C. E. Tinney, M. N. Glauser, L. S. Ukeiley, Low-dimensional characteristics of a transonic jet. Part 1. Proper orthogonal decomposition, J. Fluid Mech. 612 (2008) 107–141.

[44] E. Caraballo, M. Samimy, J. Scott, S. Narayanan, J. DeBonis, Application of proper orthogonal decomposition to a supersonic axisymmetric jet, AIAA J. 41 (5) (2003) 866–877.

[45] B. R. Noack, P. Papas, P. A. Monkewitz, The need for a pressure-term representation in empirical Galerkin models of incompressible shear flows, J. Fluid Mech. 523 (2005) 339–365.

[46] C. E. Tinney, Proper grid resolution for the proper basis, in: 47th AIAA Aerospace Sciences Meeting, Orlando, FL, 2009.

[47] T. A. Brenner, P. G. A. Cizmas, T. J. O'Brien, R. W. Breault, Practical aspects of the implementation of proper orthogonal decomposition, in: 47th AIAA Aerospace Sciences Meeting, Orlando, FL, 2009.

[48] D. J. Lucia, P. I. King, P. S. Beran, Reduced order modeling of a two-dimensional flow with moving shocks, Comput. Fluids 32 (2003) 917–938.

[49] R. Maple, P. King, J. Wolff, P. Orkwis, Split-domain harmonic balance solutions to Burger's equation for large-amplitude disturbances, AIAA J. 41 (2) (2003) 206 – 12.

[50] T. A. Brenner, R. L. Fontenot, P. G. A. Cizmas, T. J. O'Brien, R. W. Breault, Augmented proper orthogonal decomposition for problems with moving discontinuities, Powder Technol. 203 (1, Sp. Iss. SI) (2010) 78–85.

[51] J. Borggaard, T. Iliescu, Z. Wang, Artificial viscosity proper orthogonal decomposition, Math. Comput. Model. 53 (1-2) (2011) 269 – 279.

[52] H. Gunes, Low-dimensional modeling of non-isothermal twin-jet flow, Int. Commun. Heat Mass Transf. 29 (1) (2002) 77–86.

[53] C. W. Rowley, T. Colonius, R. M. Murray, Model reduction for compressible flows using POD and Galerkin projection, Physica D 189 (2004) 115–129.

[54] O. M. G. C. Op den Camp, O. S. Verheijen, L. Huisman, A. C. R. M. Backx, Application of proper orthogonal decomposition to reduce detailed CFD models of glass furnaces and forehearths, Glass Technol.-Eur. J. Glass Sci. Technol. Part A 49 (3) (2008) 119–125.

[55] E. Samadiani, Y. Joshi, Multi-parameter model reduction in multi-scale convective systems, Int. J. Heat Mass Transf. 53 (9-10) (2010) 2193–2205.

[56] D. Alonso, A. Velazquez, J. M. Vega, Robust reduced order modeling of heat transfer in a back step flow, Int. J. Heat Mass Tran. 52 (5-6) (2009) 1149–1157.

[57] P. Mokhasi, D. Rempfer, Sequential estimation of velocity fields using episodic proper orthogonal decomposition, Physica D 237 (24) (2008) 3197–3213.

[58] M.-L. Rapun, J. M. Vega, Reduced order models based on local POD plus Galerkin projection, J. Comput. Phys. 229 (8) (2010) 3046–3063.

[59] L. Sirovich, A pattern analysis of the second Rehnquist US Supreme Court, Proc. Natl. Acad. Sci. U. S. A. 100 (13) (2003) 7432–7437.

[60] R. Everson, L. Sirovich, Karhunen-Loève procedure for gappy data, J. Opt. Soc. Am. A 12 (8) (1995) 1657–1664.

[61] S. Orsborn, P. Boatwright, J. Cagan, Identifying product shape relationships using principal component analysis, Res. Eng. Des. 18 (2008) 163–180.

[62] T. Yuan, Reduced order modeling for transport phenomena based on proper orthogonal decomposition, Master's Thesis, Texas A&M University, College Station, TX, December 2003.

[63] B. R. Richardson, A reduced-order model based on proper orthogonal decomposition for two-phase flow non-isothermal transport phenomena, Master's Thesis, Texas A&M University, College Station, TX, December 2007.

[64] R. L. Fontenot, Advances in reduced-order modeling based on proper orthogonal decomposition for single and two-phase flows, Master's Thesis, Texas A&M University, College Station, TX, December 2010.

[65] L. V. Kantorovich, V. I. Krylov, Approximate Methods of Higher Analysis, Interscience Publishers, Inc., New York, 1964.

[66] L. Sirovich, Turbulence and the dynamics of coherent structures, parts I-III, Quart. Appl. Math. XLV (3) (1987) 561–590.

[67] M. Syamlal, W. Rogers, T. J. O'Brien, MFIX documentation theory guide, Tech. Rep. DOE/METC-94/1004, DOE/METC, 1994.

[68] M. Syamlal, MFIX documentation numerical technique, Tech. Rep. DE-AC21-95MC31346, EG&G Technical Services of West Virginia, 1998.

[69] D. Gidaspow, Multiphase Flow and Fluidization, Continuum and Kinetic Theory Descriptions, Academic Press, San Diego, 1994.

[70] E. Fehlberg, Low-order classical Runge-Kutta formulas with stepsize control and their application to some heat transfer problems, Tech. Rep. NASA-TR-R-315, NASA, USA, 1969.

[71] J. D. Anderson, Computational Fluid Dynamics: The Basics with Applications, McGraw-Hill, New York, 1995.

[72] P. N. Brown, G. D. Byrne, A. C. Hindmarsh, VODE: A variable coefficient ODE solver, SIAM J. Sci. Stat. Comput. 10 (5) (1989) 1038–1051.

[73] K. S. Kim, Three-dimensional hybrid grid generator and unstructured flow solver for compressors and turbines, Ph.D. Dissertation, Texas A&M University, College Station, TX, December 2003.

[74] J. I. Gargoloff, A numerical method for fully nonlinear aeroelastic analysis, Ph.D. Dissertation, Texas A&M University, College Station, TX, May 2007.

[75] D. Liliedahl, P. G. A. Cizmas, Prediction of fluid instabilities in hole-pattern stator seals, in: 47th AIAA Aerospace Sciences Meeting, AIAA Paper 2009-0786, Orlando, FL, 2009.

[76] M. Kirby, J. P. Boris, L. Sirovich, A proper orthogonal decomposition of a simulated supersonic shear layer, Int. J. Numer. Meth. Fl. 10 (1990) 411–428.

[77] J. L. Lumley, A. Poje, Low-dimensional models for flows with density fluctuations, Phys. Fluids 9 (7) (1997) 2023–2031.

[78] L. S. Ukeiley, J. M. Seiner, S. Arunajatesan, N. Sinha, S. Dash, Low-dimensional description of resonating cavity flow, in: Fluids 2000, Denver, CO, 2000.

[79] L. S. Ukeiley, C. Kannepalli, S. Arunajatesan, Low-dimensional description of variable density flows, in: 39th AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, 2001.

[80] C. W. Rowley, T. Colonius, R.M.Murray, Dynamical models for control of cavity oscillations, in: 7th AIAA/CEAS Aeroacoustics Conference, Maastricht, The Netherlands, 2001.

[81] L. Ukeiley, C. Kannepalli, S. Arunajatesan, Development of low dimensional models for control of compressible flows, in: 41st IEEE Conference on Decision and Control, Las Vegas, NV, 2002.

[82] T. Colonius, C. W. Rowley, J. B. Freund, R. M. Murray, On the choice of norm for modeling compressible flow dynamics at reduced-order using the POD, in: 41st IEEE Conference on Decision and Control, Las Vegas, NV, 2002.

[83] M. Tutkun, P. B. V. Johansson, W. K. George, Three-component vectorial proper orthogonal decomposition of axisymmetric wake behind a disk, AIAA J. 46 (5) (2008) 1118–1134.

VITA

Thomas Andrew Brenner is a native of Ovid, Michigan. In April, 2005 he earned a Bachelor of Science in Engineering in aerospace engineering from the University of Michigan.

Tom began his doctoral studies at Texas A&M University in June of 2006. His research interests include computational fluid dynamics and reduced-order models based on proper orthogonal decomposition. Tom can be contacted by email at thomasbrenner@gmail.com or by mail at: Department of Aerospace Engineering, c/o Dr. Paul Cizmas, Texas A&M University, H.R. Bright Building, Ross Street - TAMU 3141, College Station, TX 77843-3141.