SANDIA REPORT

SAND2017-10335
Unlimited Release
Printed September 2017

Advanced Fluid Reduced Order Models for Compressible Flow

Irina Tezaur, Jeffrey Fike, Kevin Carlberg, Matthew Barone, Danielle Maddix, Erin Mussoni, Maciej Balajewicz

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia National Laboratories is a multi-mission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Approved for public release; further dissemination unlimited.

Sandia National Laboratories

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

NOTICE: This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government, nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, make any warranty, express or implied, or assume any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represent that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government, any agency thereof, or any of their contractors or subcontractors. The views and opinions expressed herein do not necessarily state or reflect those of the United States Government, any agency thereof, or any of their contractors.

Printed in the United States of America. This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone: (865) 576-8401
Facsimile: (865) 576-5728
E-Mail: reports@adonis.osti.gov
Online ordering: http://www.osti.gov/bridge

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd
Springfield, VA 22161

Telephone: (800) 553-6847
Facsimile: (703) 605-6900
E-Mail: orders@ntis.fedworld.gov
Online ordering: http://www.ntis.gov/help/ordermethods.asp?loc=7-4-0#online

SAND2017-10335
Unlimited Release
Printed September 2017

Advanced Fluid Reduced Order Models for Compressible Flow

Irina Tezaur
Extreme Scale Data Science & Analytics Department
Sandia National Laboratories
P.O. Box 969, MS 9159
Livermore, CA 94551–9159
ikalash@sandia.gov

Jeffrey Fike
Aerosciences Department
Sandia National Laboratories
P.O. Box 5800, MS 0825
Albuquerque, NM 87185–0825
jafike@sandia.gov

Kevin Carlberg
Extreme Scale Data Science & Analytics Department
Sandia National Laboratories
P.O. Box 969, MS 9159
Livermore, CA 94551–9159
ktcarlb@sandia.gov

Matthew Barone
Aerosciences Department
Sandia National Laboratories
P.O. Box 5800, MS 0825
Albuquerque, NM 87185–0825
mbarone@sandia.gov

Danielle Maddix
Institute for Computational & Mathematical Engineering
Stanford University
Stanford, CA 94305
dcmaddix@stanford.edu

Erin Mussoni
Thermal/Fluid Science & Engineering Department
Sandia National Laboratories
P.O. Box 969, MS 9957
Livermore, CA 94551–9957
eemusso@sandia.gov

Maciej Balajewicz
Department of Aerospace Engineering
University of Illinois at Urbana–Champaign
306 Talbot Laboratory, MC–236
104 South Wright Street
Urbana, Illinois, 61801
mbalajew@illinois.edu

Abstract

This report summarizes fiscal year (FY) 2017 progress towards developing and implementing within the `SPARC` in-house finite volume flow solver advanced fluid reduced order models (ROMs) for compressible captive-carriage flow problems of interest to Sandia National Laboratories for the design and qualification of nuclear weapons components. The proposed projection-based model

order reduction (MOR) approach, known as the Proper Orthogonal Decomposition (POD)/Least-Squares Petrov-Galerkin (LSPG) method, can substantially reduce the CPU-time requirement for these simulations, thereby enabling advanced analyses such as uncertainty quantification and design optimization. Following a description of the project objectives and FY17 targets, we overview briefly the POD/LSPG approach to model reduction implemented within `SPARC`. We then study the viability of these ROMs for long-time predictive simulations in the context of a two-dimensional viscous laminar cavity problem, and describe some FY17 enhancements to the proposed model reduction methodology that led to ROMs with improved predictive capabilities. Also described in this report are some FY17 efforts pursued in parallel to the primary objective of determining whether the ROMs in `SPARC` are viable for the targeted application. These include the implementation and verification of some higher-order finite volume discretization methods within `SPARC` (towards using the code to study the viability of ROMs on three-dimensional cavity problems) and a novel structure-preserving constrained POD/LSPG formulation that can improve the accuracy of projection-based reduced order models. We conclude the report by summarizing the key takeaways from our FY17 findings, and providing some perspectives for future work.

Acknowledgment

Sandia National Laboratories is a multi-mission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Contents

1 Introduction **17**

2 Overview of Proper Orthogonal Decomposition (POD)/Least-Squares Petrov-Galerkin (LSPG) approach to nonlinear model reduction **19**
- 2.1 Proper Orthogonal Decomposition (POD) . . . 19
- 2.2 Least-Squares Petrov-Galerkin (LSPG) projection . . . 20
- 2.3 Hyper-reduction . . . 21
- 2.4 Implementation in `SPARC` . . . 21

3 Investigation of ROM viability on 2D viscous laminar cavity **23**
- 3.1 2D viscous laminar cavity problem description . . . 23
- 3.2 Evaluation criteria . . . 25
- 3.3 Performance of standard LSPG ROM in `SPARC` . . . 27
- 3.4 Accuracy assessment studies towards ROMs with improved predictive capabilities 31
  - 3.4.1 Project FOM snapshots onto POD basis . . . 31
  - 3.4.2 Vary start of training window . . . 33
  - 3.4.3 Vary length of training window . . . 37
  - 3.4.4 Project FOM solution increment onto POD basis . . . 41
    - Ideal preconditioned LSPG ROMs . . . 41
    - Alternative preconditioner definitions . . . 42
    - Projected solution increment results . . . 43
  - 3.4.5 Time-step refinement . . . 52
- 3.5 Summary and recommendations . . . 67

4 Implementation and verification of high-order methods in SPARC **69**
- 4.1 Summary of flux schemes in `SPARC` . . . 69
- 4.2 Description of verification methodology . . . 69
- 4.3 Mesh convergence study: inviscid pulse test case . . . 70

5 Structure preservation in finite volume ROMs via physics-based constraints **77**
- 5.1 Least-Squares Petrov-Galerkin model reduction formulation with physics-based constraints . . . 77
  - 5.1.1 Global conservation laws . . . 77
  - 5.1.2 Clausius-Duhem inequality (second law of thermodynamics) . . . 78
  - 5.1.3 Total variation diminishing (TVD) and total variation bounded (TVB) properties . . . 80
  - 5.1.4 Rotational quantities . . . 81
  - 5.1.5 Discussion . . . 82
- 5.2 Demonstration on 1D conservation laws . . . 82

5.2.1 Burger's equation . . . 82
5.2.2 Sod's shock tube problem . . . 85
5.3 Perspectives towards `SPARC` implementation . . . 91

6 Summary and future work **93**
6.1 Summary . . . 93
6.2 Future work . . . 93

References **96**

List of Figures

1.1 Compressible captive-carry problem 17

3.1 The computational mesh used for 2D viscous laminar cavity simulations in `SPARC`. 24

3.2 The pressure time history for a point midway up the downstream wall of the cavity. Shown are results for the full-order model and reduced-order model dimensional runs. The time extent shown is that used for training the reduced-order models. . . . 28

3.3 The pressure time history for a point midway up the downstream wall of the cavity. Shown are results for the full-order model and reduced-order model non-dimensional runs. The time extent shown is that used for training the reduced-order models. 28

3.4 The pressure time history for a point midway up the downstream wall of the cavity. Shown are results for the full-order model and reduced-order model non-dimensional runs. The vertical dashed lines indicate the extent of the training data used to create the POD basis used by the reduced-order models. 29

3.5 The pressure time history for a point midway up the downstream wall of the cavity. Shown are results for the full-order model and reduced-order model non-dimensional runs. The vertical dashed lines indicate the extent of the training data used to create the POD basis used by the reduced-order models. 30

3.6 The relative error between the reduced-order models and the full-order model over the entire computational domain for the non-dimensional runs. The vertical dashed lines indicate the extent of the training data used to create the POD basis used by the reduced-order models. 30

3.7 The pressure time history for a point midway up the downstream wall of the cavity. Shown are results for the full-order model, reduced-order model using 200 modes, and the full-order model snapshots projected onto 200 modes of the POD basis. The vertical dashed lines indicate the extent of the training data used to create the POD basis. 32

3.8 The relative errors of the reduced-order models runs compared to projecting full-order model snapshots onto the POD basis. The solid lines are for the FOM snapshots projected onto the POD basis, and the dashed lines are for the reduced-order models. The vertical dashed lines indicate the extent of the training data used to create the POD basis. 32

3.9 The pressure time history for a point midway up the downstream wall of the cavity. The starting point is shifted forward in time by 1 snapshot from the nominal case. Shown are results for the full-order model, reduced-order model using 200 modes, and the full-order model snapshots projected onto 200 modes of the POD basis. The vertical dashed lines indicate the extent of the training data used to create the POD basis. 33

3.10 The pressure time history for a point midway up the downstream wall of the cavity. The starting point is shifted forward in time by 2 snapshot from the nominal case. Shown are results for the full-order model, reduced-order model using 200 modes, and the full-order model snapshots projected onto 200 modes of the POD basis. The vertical dashed lines indicate the extent of the training data used to create the POD basis. 34

3.11 The pressure time history for a point midway up the downstream wall of the cavity. The starting point is shifted backward in time by 1 snapshot from the nominal case. Shown are results for the full-order model, reduced-order model using 200 modes, and the full-order model snapshots projected onto 200 modes of the POD basis. The vertical dashed lines indicate the extent of the training data used to create the POD basis. 34

3.12 The pressure time history for a point midway up the downstream wall of the cavity. The starting point is shifted backward in time by 2 snapshots from the nominal case. Shown are results for the full-order model, reduced-order model using 200 modes, and the full-order model snapshots projected onto 200 modes of the POD basis. The vertical dashed lines indicate the extent of the training data used to create the POD basis. 35

3.13 The relative errors of the reduced-order model runs compared to projecting the full-order model snapshots onto the POD basis. The solid lines are for the FOM snapshots projected onto the POD basis, and the dashed lines are for the reduced-order models. The different colors represent different training intervals. 36

3.14 The pressure time history for a point midway up the downstream wall of the cavity. Shown are results for the full-order model and reduced-order model non-dimensional runs. The vertical dashed lines indicate the extent of the training data used to create the POD basis used by the reduced-order models. 38

3.15 The pressure time history for a point midway up the downstream wall of the cavity. The starting point is shifted forward in time by 1 snapshot from the nominal case. Shown are results for the full-order model and reduced-order model runs. The vertical dashed lines indicate the extent of the training data used to create the POD basis. 38

3.16 The pressure time history for a point midway up the downstream wall of the cavity. The starting point is shifted forward in time by 2 snapshot from the nominal case. Shown are results for the full-order model and reduced-order model runs. The vertical dashed lines indicate the extent of the training data used to create the POD basis. 39

3.17 The pressure time history for a point midway up the downstream wall of the cavity. The starting point is shifted backward in time by 1 snapshot from the nominal case. Shown are results for the full-order model and reduced-order model runs. The vertical dashed lines indicate the extent of the training data used to create the POD basis. 39

3.18 The pressure time history for a point midway up the downstream wall of the cavity. The starting point is shifted backward in time by 2 snapshots from the nominal case. Shown are results for the full-order model and reduced-order model runs. The vertical dashed lines indicate the extent of the training data used to create the POD basis. 40

3.19 The pressure time history for a point midway up the downstream wall of the cavity. Shown are results for the full-order model, reduced-order model, and projected solution increment runs using an untruncated POD basis of 2000 modes. The vertical dashed lines indicate the extent of the training data used to create the POD basis. . . 44

3.20 The pressure time history for a point midway up the downstream wall of the cavity. The starting point is shifted forward in time by 1 snapshot from the nominal case. Shown are results for the full-order model, reduced-order model, and projected solution increment runs. The vertical dashed lines indicate the extent of the training data used to create the POD basis. 44

3.21 The pressure time history for a point midway up the downstream wall of the cavity. The starting point is shifted forward in time by 2 snapshot from the nominal case. Shown are results for the full-order model, reduced-order model, and projected solution increment runs. The vertical dashed lines indicate the extent of the training data used to create the POD basis. 45

3.22 The pressure time history for a point midway up the downstream wall of the cavity. The starting point is shifted backward in time by 1 snapshot from the nominal case. Shown are results for the full-order model, reduced-order model, and projected solution increment runs. The vertical dashed lines indicate the extent of the training data used to create the POD basis. 45

3.23 The pressure time history for a point midway up the downstream wall of the cavity. The starting point is shifted backward in time by 2 snapshots from the nominal case. Shown are results for the full-order model, reduced-order model, and projected solution increment runs. The vertical dashed lines indicate the extent of the training data used to create the POD basis. 46

3.24 The pressure time history for a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the three projected solution increment implementations using an untruncated POD basis of 2000 modes. The vertical dashed lines indicate the extent of the training data used to create the POD basis. 47

3.25 The power spectral density of the pressure at a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the three projected solution increment implementations using an untruncated POD basis of 2000 modes. The vertical dotted lines indicate the first and second Rossiter tones. . 49

3.26 The pressure time history for a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the three projected solution increment implementations using an untruncated POD basis of 4000 modes. The vertical dashed lines indicate the extent of the training data used to create the POD basis. 50

3.27 The power spectral density of the pressure at a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the three projected solution increment implementations using an untruncated POD basis of 4000 modes. The vertical dotted lines indicate the first and second Rossiter tones. . 51

3.28 The pressure time history and resulting power spectral at a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the standard projected solution increment implementation using an untruncated POD basis of 2000 modes. The projected solution increment case is run using 2× the nominal time step. The vertical dashed lines on the time history plot indicate the extent of the training data used to create the POD basis. The vertical dotted lines on the PSD plot indicate the first and second Rossiter tones. 53

3.29 The pressure time history and resulting power spectral at a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the standard projected solution increment implementation using an untruncated POD basis of 2000 modes. The projected solution increment case is run using 5× the nominal time step. The vertical dashed lines on the time history plot indicate the extent of the training data used to create the POD basis. The vertical dotted lines on the PSD plot indicate the first and second Rossiter tones. 54

3.30 The pressure time history and resulting power spectral at a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the standard projected solution increment implementation using an untruncated POD basis of 2000 modes. The projected solution increment case is run using 10× the nominal time step. The vertical dashed lines on the time history plot indicate the extent of the training data used to create the POD basis. The vertical dotted lines on the PSD plot indicate the first and second Rossiter tones. 55

3.31 The pressure time history and resulting power spectral at a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the standard projected solution increment implementation using an untruncated POD basis of 2000 modes. The projected solution increment case is run using 20× the nominal time step. The vertical dashed lines on the time history plot indicate the extent of the training data used to create the POD basis. The vertical dotted lines on the PSD plot indicate the first and second Rossiter tones. 56

3.32 The pressure time history and resulting power spectral at a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the standard projected solution increment implementation using an untruncated POD basis of 2000 modes. The projected solution increment case is run using 50× the nominal time step. The vertical dashed lines on the time history plot indicate the extent of the training data used to create the POD basis. The vertical dotted lines on the PSD plot indicate the first and second Rossiter tones. 57

3.33 The pressure time history and resulting power spectral at a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the second alternative projected solution increment implementation using an untruncated POD basis of 2000 modes. The projected solution increment case is run using 100× the nominal time step. The vertical dashed lines on the time history plot indicate the extent of the training data used to create the POD basis. The vertical dotted lines on the PSD plot indicate the first and second Rossiter tones. . . 58

3.34 The pressure time history and resulting power spectral at a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the second alternative projected solution increment implementation using an untruncated POD basis of 2000 modes. The projected solution increment case is run using 2× the nominal time step. The vertical dashed lines on the time history plot indicate the extent of the training data used to create the POD basis. The vertical dotted lines on the PSD plot indicate the first and second Rossiter tones. 59

3.35 The pressure time history and resulting power spectral at a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the second alternative projected solution increment implementation using an untruncated POD basis of 2000 modes. The projected solution increment case is run using 5× the nominal time step. The vertical dashed lines on the time history plot indicate the extent of the training data used to create the POD basis. The vertical dotted lines on the PSD plot indicate the first and second Rossiter tones. 60

3.36 The pressure time history and resulting power spectral at a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the second alternative projected solution increment implementation using an untruncated POD basis of 2000 modes. The projected solution increment case is run using 10× the nominal time step. The vertical dashed lines on the time history plot indicate the extent of the training data used to create the POD basis. The vertical dotted lines on the PSD plot indicate the first and second Rossiter tones. 61

3.37 The pressure time history and resulting power spectral at a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the second alternative projected solution increment implementation using an untruncated POD basis of 2000 modes. The projected solution increment case is run using 20× the nominal time step. The vertical dashed lines on the time history plot indicate the extent of the training data used to create the POD basis. The vertical dotted lines on the PSD plot indicate the first and second Rossiter tones. 62

3.38 The pressure time history and resulting power spectral at a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the second alternative projected solution increment implementation using an untruncated POD basis of 2000 modes. The projected solution increment case is run using 50× the nominal time step. The vertical dashed lines on the time history plot indicate the extent of the training data used to create the POD basis. The vertical dotted lines on the PSD plot indicate the first and second Rossiter tones. 63

3.39 The pressure time history and resulting power spectral at a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the second alternative projected solution increment implementation using an untruncated POD basis of 2000 modes. The projected solution increment case is run using 100× the nominal time step. The vertical dashed lines on the time history plot indicate the extent of the training data used to create the POD basis. The vertical dotted lines on the PSD plot indicate the first and second Rossiter tones. . . 64
3.40 The pressure time history and resulting power spectral at a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the standard projected solution increment implementation using an untruncated POD basis of 4000 modes. The projected solution increment case is run using 2× the nominal time step. The vertical dashed lines on the time history plot indicate the extent of the training data used to create the POD basis. The vertical dotted lines on the PSD plot indicate the first and second Rossiter tones. 66

4.1 The computed relative error between the solution on each mesh and the solution on the finest mesh for the inviscid pulse test case. The results shown are for the second-order Roe flux scheme in `SPARC`. 71
4.2 The computed relative error as a function of mesh size for the inviscid pulse test case at 34 seconds. The results shown are for the second-order Roe flux scheme in `SPARC`. The convergence rate is close to 2, as expected from theory. 71
4.3 The computed convergence rate between mesh sizes as a function of time for the inviscid pulse test case. The results shown are for the second-order Roe flux scheme in `SPARC`. The blue and red lines correspond to the slopes of the right and left segment of the line in Figure 4.2, respectively. As the mesh is refined, the asymptotic convergence rate of 2 is achieved. 72
4.4 The relative errors between the solution on each mesh and the solution on the finest mesh for the inviscid pulse test case. The results shown are for the first-order Roe flux scheme in `SPARC`. 73
4.5 The relative errors between the solution on each mesh and the solution on the finest mesh for the inviscid pulse test case. The results shown are for the second-order Roe flux scheme in `SPARC`. 73
4.6 The relative errors between the solution on each mesh and the solution on the finest mesh for the inviscid pulse test case. The results shown are for the second-order Steger-Warming flux scheme in `SPARC`. 74
4.7 The relative errors between the solution on each mesh and the solution on the finest mesh for the inviscid pulse test case. The results shown are for the second-order Subbareddy-Candler flux scheme in `SPARC`. 74
4.8 The relative errors between the solution on each mesh and the solution on the finest mesh for the inviscid pulse test case. The results shown are for the fourth-order Subbareddy-Candler flux scheme in `SPARC`. 75
4.9 The relative errors for each flux scheme as the mesh size is varied. 76
4.10 The convergence rates as a function of time for each flux scheme. 76

5.1 Spatial profile as the solution evolves with time. 83

5.2 Spatial profile as the shock forms and evolves with time. 83
5.3 Conserved variable and total variation as functions of time. 84
5.4 Relative state-space error and total variation as functions of time. 84
5.5 Spatial profile at time $t_{ROMfinal} = 0.1$. 86
5.6 Spatial profile at time $t_{ROMfinal} = 0.2$. 87
5.7 Spatial profile at time $t_{ROMfinal} = 0.3$. 88
5.8 Spatial profile at time $t_{ROMfinal} = 0.4$. 89
5.9 Total conserved variables as a function of time. 89
5.10 Total variation as a function of time 90
5.11 Relative state space error as a function of time. 90

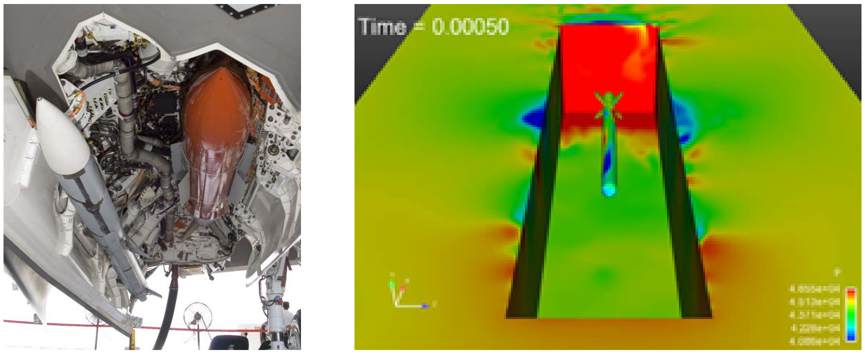
List of Tables

3.1 Parameters used for the 2D viscous laminar cavity test case in `SPARC` 24

Chapter 1

Introduction

The aim of the project described in this report is to develop and implement reduced order models (ROMs) of the aircraft weapons bay (also known as "captive carriage") environment (Figure 1.1), a compressible fluid flow application of immediate interest to Sandia National Laboratories for the design and qualification of nuclear weapons (NW) systems. Existing in-house Large Eddy Simulation (LES) and Direct Numerical Simulation (DNS) Computational Fluid Dynamics (CFD) codes, e.g., the `Sigma CFD` [1] and `SPARC` [3] flow solvers, are able to simulate the captive carriage environment to a sufficient accuracy, but require very fine meshes and long run-times. The fact that a single relevant captive-carry simulation can take on the order of *weeks* makes advanced analyses requiring numerous simulations, e.g., uncertainty quantification (UQ) and component design optimization, intractable. Projection-based reduced order models, models constructed from one or more high-fidelity simulations that retain the essential physics and dynamics of their corresponding full order models (FOMs) but have a much smaller computational cost, are a promising tool for alleviating this difficulty. Projection-based model order reduction (MOR) for nonlinear problems has three main steps: (1) calculation of a low-dimensional subspace through a data compression of a set of snapshots collected from a high-fidelity simulation, (2) projection of the governing partial differential equations (PDEs) or full order model (FOM) onto this low-dimensional subspace, and (3) hyper-reduction, an approach for efficiently computing the projection of nonlinear terms in the governing PDEs or FOM by effectively evaluating these terms at a small number of carefully-selected points. The result of this procedure is a small set of time-dependent ordinary differential equations (ODEs) for the ROM modal amplitudes that approximately describe the flow dynamics of a FOM for some limited set of flow conditions.



(a) Airplane weapons bay **(b)** Compressible cavity with object

Figure 1.1. Compressible captive-carry problem

This report summarizes fiscal year (FY) 2017 progress towards an ongoing research effort to develop and implement within the `SPARC` in-house flow solver stable, accurate, robust and efficient projection-based ROMs for compressible captive-carriage problems. These ROMs are based on the Proper Orthogonal Decomposition (POD)/Least-Squares Petrov-Galerkin (LSPG) approach, a minimal-residual-based method that creates the ROM from a fully-discrete high-fidelity CFD discretization. When combined with a hyper-reduction approach known as gappy POD [14], this method is equivalent to the Gauss-Newton with Approximated Tensors (GNAT) method of Carlberg *et al.* [8]. This methodology and its implementation within `SPARC` is described in detail in this project's FY16 report, [45].

During FY16, the viability of Galerkin and minimal-residual projection-based model reduction algorithms implemented within the `SPARC` CFD research code was investigated on some laminar cavity flow problems at moderate Reynolds number. It was demonstrated that, while the SPARC ROMs were usually capable of reproducing a training data set, apparent instabilities precluded long-time predictive simulations using the ROMs. Much of the FY17 effort, summarized in the present document, has focused on resolving these computational difficulties. A secondary FY17 effort involved the creation of a code verification methodology and workflow, towards enabling the implementation and verification of higher-order discretization methods (e.g., the Rai scheme [35]) in `SPARC`. Although seemingly irrelevant to `SPARC` MOR development, higher-order FOM discretizations in `SPARC` are necessary for three-dimensional (3D) cavity ROM simulations using this code base at a later point in time.

Toward this effect, the remainder of this report is organized as follows. In Chapter 2, the POD/LSPG approach to nonlinear model reduction is reviewed succinctly to keep this document self-contained. In Chapter 3, we study the viability of the POD/LSPG ROMs in `SPARC` for long-time predictive simulations in the context of a two-dimensional (2D) viscous laminar cavity problem. After describing the problem formulation and our evaluation criteria, which center around the ROMs' ability to capture flow statistics (e.g., pressure power spectral densities, or PSDs) for long-time simulations, we discuss some FY17 enhancements to our ROM implementation and methodology that led to ROMs with better predictive capabilities. These improvements include the addition of a capability to run `SPARC` ROMs non-dimensionally to improve these models' numerical performance, as well as the discovery of preconditioners that have the effect of modifying the norm in which the residual is minimized and yield ROMs with improved accuracy for long-time predictive runs. Chapter 4 details the implementation and verification of some higher-order finite volume discretization methods within `SPARC`. Towards defining future directions, Chapter 5 highlights some key aspects of a parallel research effort undertaken during FY17 involving the development of a structure-preserving constrained POD/LSPG model reduction formulation. Finally, we summarize and discuss the key takeaways from our FY17 findings and give some perspectives for future work in Chapter 6.

Chapter 2

Overview of Proper Orthogonal Decomposition (POD)/Least-Squares Petrov-Galerkin (LSPG) approach to nonlinear model reduction

To keep this report self-contained, we succinctly overview the POD/Least-Squares Petrov-Galerkin approach to model reduction in this chapter. This approach was implemented within the `SPARC` flow solver during FY15-FY16.

Consider the following system of nonlinear equations

$$\boldsymbol{r}(\boldsymbol{w}) = \boldsymbol{0} \tag{2.1}$$

where $\boldsymbol{w} \in \mathbb{R}^N$ is the state vector and $\boldsymbol{r} : \mathbb{R}^N \to \mathbb{R}^N$ is the nonlinear residual operator. In this case, (2.1) are the compressible Navier-Stokes equations, discretized in space and time, so that (2.1) is the (discrete) full order model (FOM) for which we will build a ROM. Assuming (2.1) is solved using a (globalized) Newton's method, the sequence of solutions generated are

$$\boldsymbol{J}^{(k)} \delta \boldsymbol{w}^{(k)} = -\boldsymbol{r}^{(k)}, \quad k = 1, \ldots, K \tag{2.2}$$

$$\boldsymbol{w}^{(k)} = \boldsymbol{w}^{(k-1)} + \alpha_k \delta \boldsymbol{w}^{(k)}, \tag{2.3}$$

where $\boldsymbol{J}^{(k)} := \frac{\partial \boldsymbol{r}}{\partial \boldsymbol{w}} \left(\boldsymbol{w}^{(k)} \right) \in \mathbb{R}^{N \times N}$, $\boldsymbol{r}^{(k)} := \boldsymbol{r} \left(\boldsymbol{w}^{(k)} \right) \in \mathbb{R}^N$, $\boldsymbol{w}^{(0)}$ is an initial guess for the solution, and $\alpha_k \in \mathbb{R}$ is the step length (often set to one).

2.1 Proper Orthogonal Decomposition (POD)

The first step in the LSPG/POD approach to model reduction is the calculation of a basis of reduced dimension $M << N$ (where N denotes the number of degrees of freedom in the full order model (2.1)) using the POD. The POD is a mathematical procedure that, given an ensemble of data and an inner product, denoted generically by $(\cdot, \cdot)$, constructs a basis for the ensemble that is optimal in the sense that it describes more energy (on average) of the ensemble in the chosen inner product than any other linear basis of the same dimension M. The ensemble $\{\mathbf{x}^k : k = 1, \ldots, K\}$ is typically a set of K instantaneous snapshots of a numerical solution field, collected for K values of a parameter of interest, or at K different times. Mathematically, POD seeks an M-dimensional ($M << K$) subspace spanned by the set $\{\boldsymbol{\phi}_i\}$ such that the projection of the difference between the ensemble $\mathbf{x}^k$ and its projection onto the reduced subspace is minimized on average. It is a well-known result [4, 21, 26, 36] that the solution to the POD optimization problem reduces to the eigenvalue problem

$$\mathbf{R}\boldsymbol{\phi} = \lambda \boldsymbol{\phi}, \tag{2.4}$$

where $\mathbf{R}$ is a self-adjoint and positive semi-definite operator with its (i, j) entry given by $R_{ij} = \frac{1}{K}\left(\mathbf{x}^i, \mathbf{x}^j\right)$ for $1 \leq i, j \leq K$. It can be shown [21, 29] that the set of M eigenfunctions, or POD modes, $\{\boldsymbol{\phi}_i : i = 1, \ldots, M\}$ corresponding to the M largest eigenvalues of $\mathbf{R}$ is precisely the desired basis. This is the so-called "method of snapshots" for computing a POD basis [40].

Once a reduced basis is obtained, we approximate the solution to (2.1) by

$$\tilde{\boldsymbol{w}} = \bar{\boldsymbol{w}} + \boldsymbol{\Phi}_M \hat{\boldsymbol{w}} = \bar{\boldsymbol{w}} + \sum_{i=1}^{M} \boldsymbol{\phi}_i \hat{w}_i \tag{2.5}$$

with $\hat{\boldsymbol{w}} := [\hat{w}_1 \ \cdots \ \hat{w}_M]^T \in \mathbb{R}^M$ denoting the generalized coordinates, and $\bar{\boldsymbol{w}} \in \mathbb{R}^N$ denoting a reference solution, often taken to be the initial condition in the case of an unsteady simulation.

We then substitute the approximation (2.5) into (2.1). This yields

$$\boldsymbol{r}(\bar{\boldsymbol{w}} + \boldsymbol{\Phi}_M \hat{\boldsymbol{w}}) = \mathbf{0}, \tag{2.6}$$

which is a system of N equations in M unknowns $\hat{\boldsymbol{w}}$. As this is an over-determined system, it may not have a solution.

2.2 Least-Squares Petrov-Galerkin (LSPG) projection

In the LSPG approach to model reduction, solving the ROM for (2.1) amounts to solving the following least-squares optimization problem

$$\hat{\boldsymbol{w}}_{\text{PG}} = \arg \min_{\boldsymbol{y} \in \mathbb{R}^M} \|\boldsymbol{r}(\bar{\boldsymbol{w}} + \boldsymbol{\Phi}_M \boldsymbol{y})\|_2^2. \tag{2.7}$$

Here, the approximate solution is $\tilde{\boldsymbol{w}}_{\text{PG}} := \bar{\boldsymbol{w}} + \boldsymbol{\Phi}_M \hat{\boldsymbol{w}}_{\text{PG}}$. The name "LSPG" ROM comes from the observation that solving (2.7) amounts to solving a nonlinear least-squares problem. The two most popular approaches for this are the Gauss–Newton approach and the Levenberg–Marquardt (trust-region) method. Following the work of Carlberg *et al.* [8], we adopt the Gauss–Newton approach[1].

---

[1] The LSPG approach is the basis for the Gauss–Newton with Approximated Tensors (GNAT) method of Carlberg *et al.* [8].

This approach implies solving a sequence of linear least-squares problems of the form

$$\delta\hat{\boldsymbol{w}}_{\mathrm{PG}}^{(k)} = \arg\min_{\boldsymbol{y}\in\mathbb{R}^M} \|\boldsymbol{J}^{(k)}\boldsymbol{\Phi}_M\boldsymbol{y} + \boldsymbol{r}^{(k)}\|_2^2, \quad k = 1,\ldots,K_{\mathrm{PG}} \tag{2.8}$$

$$\hat{\boldsymbol{w}}_{\mathrm{PG}}^{(k)} = \hat{\boldsymbol{w}}_{\mathrm{PG}}^{(k-1)} + \alpha_k\delta\hat{\boldsymbol{w}}_{\mathrm{PG}}^{(k)} \tag{2.9}$$

$$\tilde{\boldsymbol{w}}_{\mathrm{PG}}^{(k)} = \bar{\boldsymbol{w}} + \boldsymbol{\Phi}_M\hat{\boldsymbol{w}}_{\mathrm{PG}}^{(k-1)}, \tag{2.10}$$

where K_{PG} is the number of Gauss-Newton iterations. It can be shown that the approximation upon convergence is $\tilde{\boldsymbol{w}}_{\mathrm{PG}} = \tilde{\boldsymbol{w}}_{\mathrm{PG}}^{(K_{\mathrm{PG}})}$ and $\hat{\boldsymbol{w}}_{\mathrm{PG}} = \hat{\boldsymbol{w}}_{\mathrm{PG}}^{(K)}$.[2] Note that the normal-equations form of (2.8) is

$$\boldsymbol{\Phi}_M^T\boldsymbol{J}^{(k)T}\boldsymbol{J}^{(k)}\boldsymbol{\Phi}_M\delta\hat{\boldsymbol{w}}_{\mathrm{PG}}^{(k)} = -\boldsymbol{\Phi}_M^T\boldsymbol{J}^{(k)T}\boldsymbol{r}^{(k)}, \quad k = 1,\ldots,K_{\mathrm{PG}}, \tag{2.11}$$

which can be interpreted as a Petrov–Galerkin process of the Newton iteration with trial basis (in matrix form) $\boldsymbol{\Phi}_M$ and test basis $\boldsymbol{J}^{(k)}\boldsymbol{\Phi}_M$.

The simplest implementation of the Gauss–Newton method for solving (2.7). For details of this approach and its implementation in `SPARC`, the reader is referred to Chapters 2 and 3 of [45], respectively.

2.3 Hyper-reduction

The LSPG projection approach described in Section 2.2 is efficient for nonlinear problems. This is because the solution of the ROM system requires algebraic operations that scale with the dimension of the original full-order model N. This problem can be circumvented through the use of hyper-reduction. A number of hyper-reduction approaches have been proposed, including the discrete empirical interpolation method (DEIM) [11], “best points” interpolation [31, 32], collocation [27] and gappy POD [14]. Implementation of the latter two approaches has been started within `SPARC`, but is not complete at the present time, and hence a detailed discussion of this methodology goes beyond the scope of this report. The basic idea behind these approaches is to compute the residual at some small number of points q with $q << N$, encapsulated in a “sampling matrix” $\boldsymbol{Z}$. This set of q points is typically referred to as the “sample mesh”. The “sample mesh” is computed by solving an optimization problem offline; see Section 2.3 of [45] and [8] for details. The LSPG projection approach combined with gappy POD hyper-reduction is equivalent to the GNAT method [8].

2.4 Implementation in `SPARC`

The POD/LSPG method described above has been implemented in the `SPARC` compressible flow solver. For details of this implementation, the reader is referred to Chapter 3 of [45].

---

[2] In the event of an unsteady simulation, the initial guess for the generalized coordinates is taken to be the generalized coordinates at the previous time step.

Chapter 3

Investigation of ROM viability on 2D viscous laminar cavity

3.1 2D viscous laminar cavity problem description

The LSPG/POD ROMs implemented in `SPARC` are evaluated on a test case involving a 2D viscous laminar flow around an open cavity geometry, described in this subsection.

The computational domain for this test case is composed of two rectangular regions: a cavity region, $\Omega_{cavity} = [0.0m, 0.0917136m] \times [0.0m, -0.0458568m]$, and an outer flow region, $\Omega_{flow} = [-4.58568m, 4.58568m] \times [0.0m, 6.87852m]$. This 2D domain is made into a 3D domain for `SPARC` by making the mesh 1 cell thick and imposing symmetry (inviscid, slip wall) boundary conditions on the faces parallel to the plane. The nominal mesh used for these simulations contains 104,500 hexahedral cells and is shown in Figure 3.1. A formal mesh convergence study for this geometry is planned but has not been completed yet[1]. The large extent of the outer flow domain is intended to minimize the effects of any pressure waves reflecting off the boundaries. Reflections off the boundaries were seen to have a significant impact on the accuracy and stability of ROMs in previous work [24]. Previous runs using other codes, such as `Sigma CFD`, utilized a sponge region to eliminate reflected pressure waves. As `SPARC` does not currently have a sponge boundary condition implemented, the outer domain was made very large and the cells stretched in the far field in order to minimize any pressure wave reflections.

The flow conditions for this test case are chosen to produce approximately Mach 0.6 and a Reynolds number of approximately 3000. The exact parameters specified in the `SPARC` input file are given in Table 3.1. The nominal full-order model runs used BDF2 time stepping with a fixed time step corresponding to a CFL number of under 50.0. The time step was chosen in order to limit the number of snapshots that are produced during the ROM training interval.

Viscous, no-slip boundary conditions are imposed on the left, right, and bottom surfaces of the cavity domain, Ω_{cavity}. Far-field boundary conditions are imposed on the left, right, and top surfaces of the outer flow domain, Ω_{flow}. A combination of inviscid slip wall and viscous no-slip wall boundary conditions are imposed on the lower boundary of Ω_{flow}. The regions immediately before and after the cavity have no-slip walls, but the regions closer to the inflow and outflow surfaces

---

[1] See Chapter 4 for some `SPARC` mesh convergence studies in the context of a simplified box geometry.

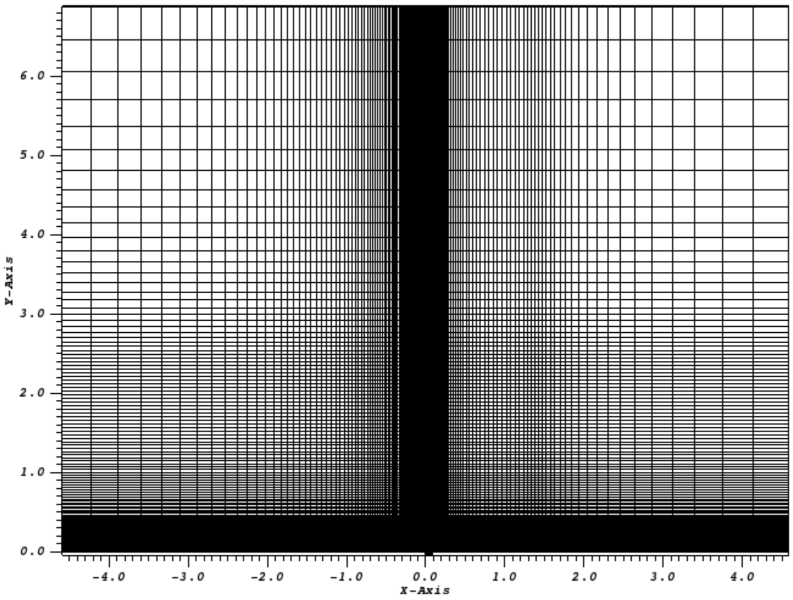


Figure 3.1. The computational mesh used for 2D viscous laminar cavity simulations in `SPARC`.

Parameter	Dimensional Value	Non-Dimensional Value
Free-stream Velocity, u_1	$208.7816\, m/s$	0.601247049354
Temperature, T	$300\, K$	1.0
Density, ρ	$2.9026155498083859 \times 10^{-4}\, kg/m^3$	1.4
Pressure, p	$25.0\, Pa$	1.0
Viscosity, μ	$8.46 \times 10^{-7}\, kg/(m\, s)$	$1.17508590713 \times 10^{-5}$
Specific Gas Constant, R	$287.097384766765\, m^2/(s^2\, K)$	0.714285714286
γ	1.4	1.4
Prandtl Number	0.72	0.72
Time Step	$2.0 \times 10^{-6}\, s$	$6.9449521698 \times 10^{-4}$

Table 3.1. Parameters used for the 2D viscous laminar cavity test case in `SPARC`.

have inviscid slip walls. This strategy allows constant far-field inflow conditions to be specified without having to impose a boundary layer profile. The boundary layer begins to grow at the upstream transition from inviscid wall to no-slip wall. The extent of the no-slip wall was chosen to allow the boundary layer to attain the desired thickness at the beginning of the cavity.

Reduced-order models are created and run in `SPARC` in several stages. First, the full-order model in `SPARC` is run for 100,000 time steps with a 10× smaller time step than that given in Table 3.1. This initial run is performed to advance the simulation past any initial transients and reach a somewhat periodic state. The full-order model is then restarted from a point past the initial transient behavior using the time step given in Table 3.1 and snapshots of the flow field are recorded at every time step. The reason for the change in time step is to reduce the number of snapshots that are recorded. Proper Orthogonal Decomposition (POD) is then performed on these snapshots to create a modal basis. This modal basis is then used when reduced-order models are run in `SPARC`.

3.2 Evaluation criteria

Reduced-order models typically trade reduced accuracy for reduced computational cost. It is therefore important to be able to assess the accuracy of the ROMs and determine if the reduced accuracy is acceptable. This can be done by running both the full-order model and reduced-order models for the same flow and comparing the results.

There are several ways to compare the results from the full-order model and reduced-order models. No method of comparison is perfect; each method has its strengths and weaknesses. A good comparison between the full-order model and the reduced-order model will likely require a combination of methods.

One way to compare two simulations is to use a flow visualization package to visualize the flow field as the solution evolves in time. This enables us to see if the overall general flow features are accurately captured by the reduced-order model. Flow field animations also allow us to see if there are discrepancies in frequency or phase between the FOM and the ROM. Discrepancies in the amplitude are a little more difficult to discern. Flow field visualizations can provide a qualitative ("eyeball norm") comparison between simulations but do not provide a quantitative measure of the error.

The error between two simulations can be computed by taking the norm of the difference between the two data sets at every point in the domain. The relative error between two flow solutions can be computed as

$$\mathscr{E} = \frac{\|q_2 - q_1\|_2}{\|q_1\|_2} \,, \tag{3.1}$$

where q is some quantity of interest. The subscript 1 is for the full-order model or true solution, and the subscript 2 is for the reduced-order model solution or whatever solution we are comparing to the true solution. Using a constant time step for all simulations guarantees that this difference can be performed at every time step. The quantity of interest, q in (3.1), could be a scalar quantity such as the pressure, or it could be a vector quantity such as the conservative solution variables. We often compute the total error between the conservative variables as

$$\mathscr{E}_{total} = \frac{\left(\sum\limits_{cells} (\rho_1 - \rho_2)^2 + ((\rho u)_1 - (\rho u)_2)^2 + ((\rho v)_1 - (\rho v)_2)^2 + ((\rho w)_1 - (\rho w)_2)^2 + ((\rho e)_1 - (\rho e)_2)^2\right)^{\frac{1}{2}}}{\left(\sum\limits_{cells} (\rho_1)^2 + (\rho u)_1^2 + (\rho v)_1^2 + (\rho w)_1^2 + (\rho e)_1^2\right)^{\frac{1}{2}}} . \tag{3.2}$$

Here, ρ denotes the fluid density; u, ,v and w are the xR–, y– and z– components of the fluid velocity respectively; e is the fluid energy. The sum in (3.2) is over the cells in the mesh. As written, this expression is sensitive to the relative magnitudes of the conservative variables and will be dominated by the variable with the largest magnitude. This sensitivity is reduced when the variables are non-dimensional and of the same magnitude.

It is often not feasible to record snapshots of the entire flow field at every time step. Therefore flow visualizations or flow field errors have lower temporal resolution than the simulations. An alternative is to output data every time step but at only a few specified locations. The use of probes or traces of this type increases the temporal resolution of the data, but there is much less spatial information. It is therefore important to place these probes in locations where the variations in the quantities of interest are desired. For the cavity flow, we focus on a point midway up the downstream wall of the cavity. Other locations of interest may be on the floor of the cavity or in the shear layer at the top of the cavity.

In `SPARC`, probes output the conservative variables at each time step. For the cavity flow, the primary quantity of interest are the pressure fluctuations, so the conservative variables are post-processed to produce a pressure time history. Plotting the pressure time histories from both the FOM and the ROM enables a comparison of the amplitude, phase, and frequency of oscillations.

It is not anticipated that the ROM will be able to exactly reproduce the pressure time history from the FOM. Small differences in the flow state can produce substantially different time histories as the simulations are run for long time periods. However, the statistics of these time histories, such as the root-mean-square (rms) value or the power spectral density (PSD), should be similar.

The power spectral density is computed by taking the magnitude of the Fourier transform of the time signal. This provides information on the power of the oscillations at each frequency in the time history. The area under the PSD curve is the rms value of the oscillations, and provides information on the average amplitude of the oscillations.

None of the comparison methods discussed above are perfect. A combination of the methods should be used to provide a sufficient comparison between the ROM and the FOM. Using probes we can obtain pressure time histories at one or more points. PSDs can then be computed at these points. Plots comparing the pressure time histories and PSDs from different simulations provide a good means of comparing the two simulations. However, these comparisons are localized to the probe locations. Flow visualizations of the entire flow field provide a more global comparison.

Plotting time histories or PSD curves and visualizing the entire flow field provides a qualitative comparison between two simulations. A quantitative measure of the error can be computed using (3.1) on either the full flow field, the pressure time histories, or the PSDs. However, there can be situations where the solutions are qualitatively similar but have a large computed error. Small shifts in frequency or phase can produce large errors between flow fields or time histories. Small shifts in the frequencies of the peaks on the PSD can also produce large error values.

Quantitative measures of the error are useful to be able to assess how much of an effect changes to the ROMs, such as the number of modes, have on the accuracy of the simulations. However, these quantitative measures should be combined with qualitative comparisons to provide a complete assessment of the difference between two simulations.

3.3 Performance of standard LSPG ROM in `SPARC`

During FY16, ROM capabilities were added to `SPARC` and tested on both the 2D viscous laminar cavity discussed above and an inviscid pressure pulse test case. For the 2D cavity runs, several issues were identified, including the inability of the ROM to accurately reproduce the behavior of the FOM in the training interval, even when all the modes were retained. The exact cause of this behavior is unknown. Upon updating `SPARC` to a newer version, the ROM was able to better reproduce the FOM. It was also determined that the default convergence criteria in `SPARC` should be tightened in order to give the best chance of producing better results. Results for the 2D cavity generated during FY17 reflect both of these changes. Figure 3.2 shows the pressure time history at a point midway up the downstream cavity wall. This figure compares the full-order model with reduced-order models using 100 or 200 modes. As can be seen, the reduced order models can accurately reproduce the full order model behavior during the training interval. This figure can be contrasted to Figure 5.6 or 5.10 in our FY16 report.

The inviscid pressure pulse test case was used to help investigate the ROM behavior on a smaller problem. For the inviscid pressure pulse case, it was found that running the FOM and ROM non-dimensionally dramatically improved the ROM performance. However, non-dimensionalization seemed to have had little effect on the 2D cavity. Figure 3.3 shows results for the non-dimensional full-order model and reduced-order model runs. The reduced-order models are able to reproduce the full-order model results during the training window. Based on these results, there does not appear to be a benefit to running the cavity simulations non-dimensionally. Despite this, it was decided that cavity runs should primarily be performed non-dimensionally in order to provide the best chance of success.

The above results provide confidence that the reduced-order model implementation is functioning correctly. However, the goal of this work is to use reduced-order models in situations other than simply reproducing the training data. In particular, we are interested in running the ROM further in time, beyond the end of the training window. Figures 3.4 and 3.5 show a comparison of the full-order model and reduced-order model results as the simulations are run beyond the end of the training window. These plots show similar results for both the dimensional and non-dimensional runs. The reduced-order models accurately reproduce the full-order model during the training win-

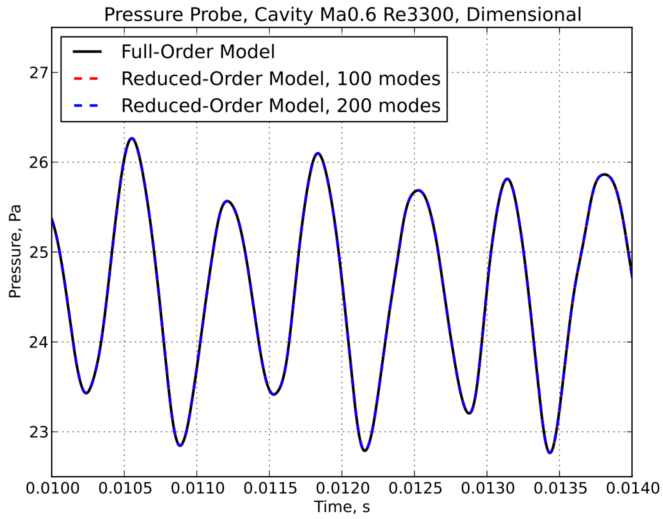


Figure 3.2. The pressure time history for a point midway up the downstream wall of the cavity. Shown are results for the full-order model and reduced-order model dimensional runs. The time extent shown is that used for training the reduced-order models.

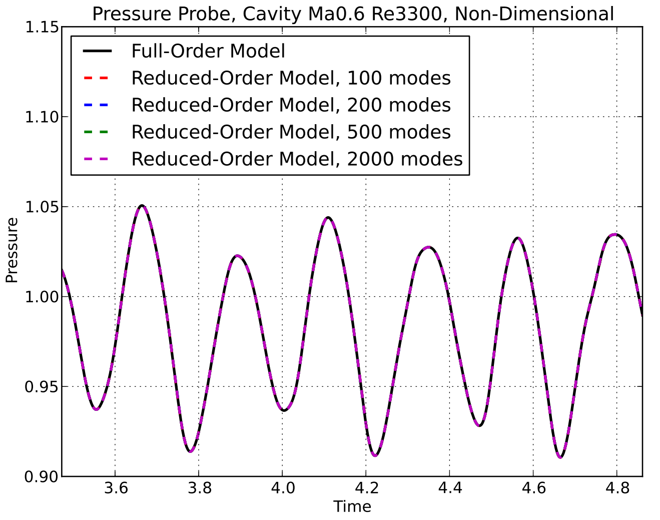


Figure 3.3. The pressure time history for a point midway up the downstream wall of the cavity. Shown are results for the full-order model and reduced-order model non-dimensional runs. The time extent shown is that used for training the reduced-order models.

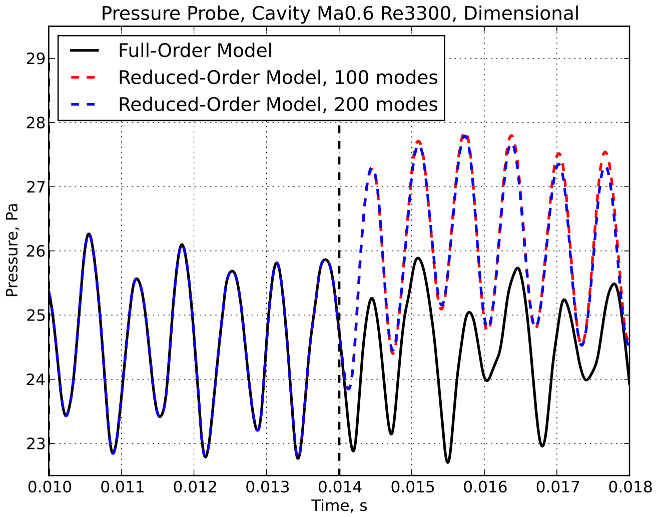


Figure 3.4. The pressure time history for a point midway up the downstream wall of the cavity. Shown are results for the full-order model and reduced-order model non-dimensional runs. The vertical dashed lines indicate the extent of the training data used to create the POD basis used by the reduced-order models.

dow, but there are significant errors once the simulations advance further in time. Figure 3.6 shows the error between the reduced-order models and the full-order model for the non-dimensional runs computed using (3.2). This plot reinforces the observation that the reduced-order model is accurate during the training window, but exhibits significant error after the end of the training window. This plot also seems to show that using an increased number of modes increases the accuracy during the training interval, but beyond the end of the training interval the number of modes used does not appear to have a significant effect on the accuracy. However, this observation may be due to the nature of the error calculation and may change if an alternative measure of accuracy is used.

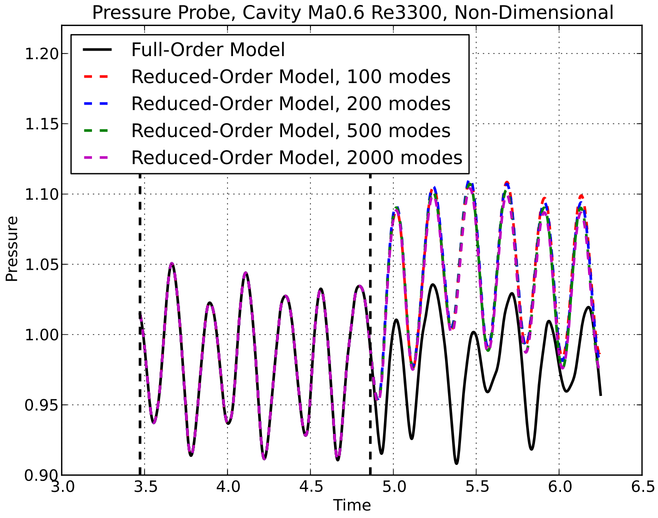


Figure 3.5. The pressure time history for a point midway up the downstream wall of the cavity. Shown are results for the full-order model and reduced-order model non-dimensional runs. The vertical dashed lines indicate the extent of the training data used to create the POD basis used by the reduced-order models.

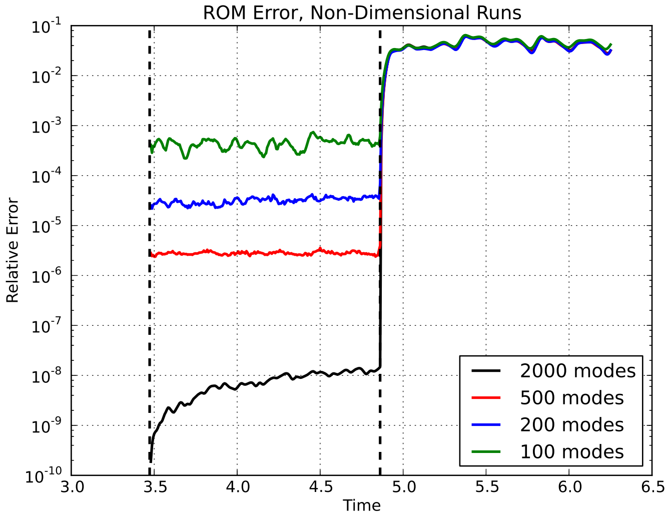


Figure 3.6. The relative error between the reduced-order models and the full-order model over the entire computational domain for the non-dimensional runs. The vertical dashed lines indicate the extent of the training data used to create the POD basis used by the reduced-order models.

3.4 Accuracy assessment studies towards ROMs with improved predictive capabilities

We undertook several studies to try to assess and improve the performance of the `SPARC` ROMs for long-time predictive simulations, which are summarized in this section.

3.4.1 Project FOM snapshots onto POD basis

One question which arises is whether the POD basis constructed using the training data is capable of accurately representing the flow after the end of the training interval. This was assessed by projecting the snapshots from the FOM run onto the POD basis to create a set of projection coefficients, then using the basis and these projection coefficients to construct an approximation of the flow field. If $\boldsymbol{w}^{(k)}$ is the snapshot of the conservative variables at time instance k from the FOM, then the approximation to the flow field, $\tilde{\boldsymbol{w}}^{(k)}$, is given by

$$\tilde{\boldsymbol{w}}^{(k)} = \bar{\boldsymbol{w}} + \boldsymbol{\Phi}_M \left(\boldsymbol{\Phi}_M^T \left(\boldsymbol{w}^{(k)} - \bar{\boldsymbol{w}} \right) \right) , \tag{3.3}$$

where $\boldsymbol{\Phi}_M$ is the POD basis and $\bar{\boldsymbol{w}}$ is some base flow state. This approximated flow state is the best that can be constructed using a given number of modes from the POD basis.

Figure 3.7 shows the pressure time history that results from this approximated flow field when 200 modes are used. Also shown are results for the true full-order model and the reduced-order model using 200 modes. The approximated solution reproduces the full-order model results fairly accurately, even beyond the end of the training interval. Figure 3.8 shows the errors of both the ROM and FOM projected onto the basis as the number of modes is varied. In this figure the solid lines are for the FOM projected onto the basis, and the dashed lines are for the reduced-order models. This plot shows that even though the pressure time history for the FOM projected onto the basis looked fairly accurate throughout the run, there is still a large increase in error after the end of the training interval.

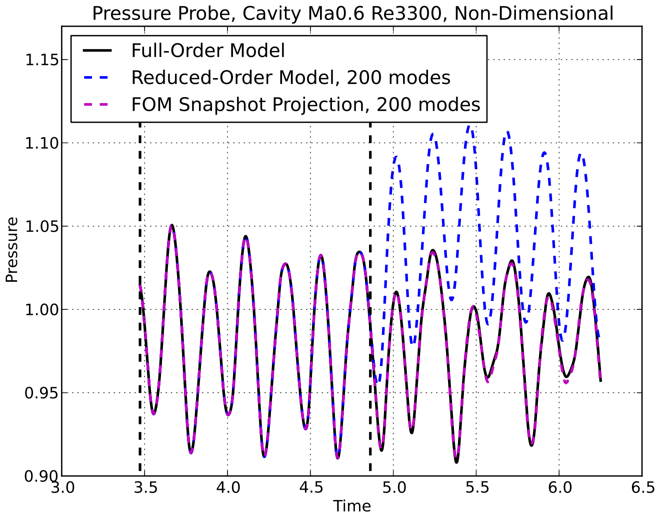


Figure 3.7. The pressure time history for a point midway up the downstream wall of the cavity. Shown are results for the full-order model, reduced-order model using 200 modes, and the full-order model snapshots projected onto 200 modes of the POD basis. The vertical dashed lines indicate the extent of the training data used to create the POD basis.

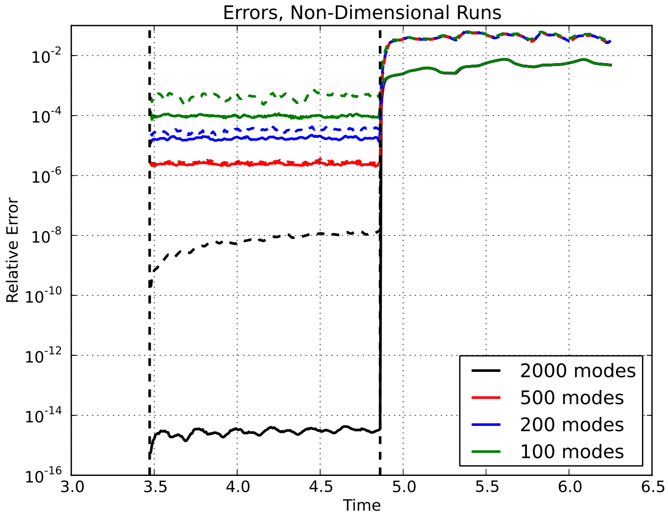


Figure 3.8. The relative errors of the reduced-order models runs compared to projecting full-order model snapshots onto the POD basis. The solid lines are for the FOM snapshots projected onto the POD basis, and the dashed lines are for the reduced-order models. The vertical dashed lines indicate the extent of the training data used to create the POD basis.

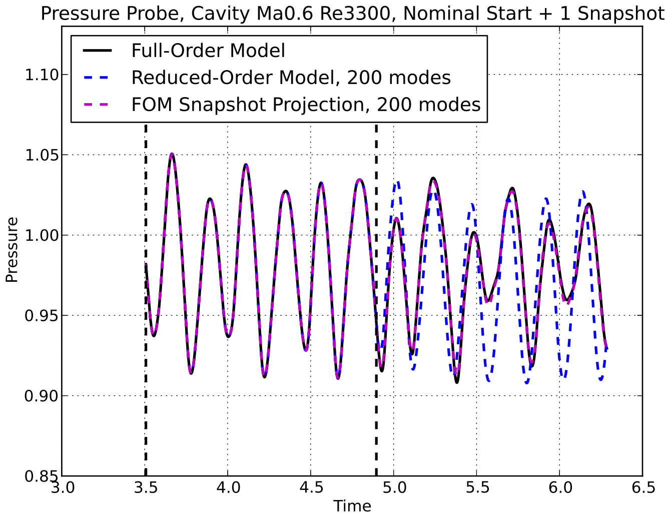


Figure 3.9. The pressure time history for a point midway up the downstream wall of the cavity. The starting point is shifted forward in time by 1 snapshot from the nominal case. Shown are results for the full-order model, reduced-order model using 200 modes, and the full-order model snapshots projected onto 200 modes of the POD basis. The vertical dashed lines indicate the extent of the training data used to create the POD basis.

3.4.2 Vary start of training window

The exact starting point of the training interval was chosen somewhat arbitrarily. Based upon examination of Figures 3.4 and 3.5, one might consider how shifting the training interval might affect the ROM performance. It is possible that if the training interval was shifted slightly forward in time, then the ROM behavior after the end of the training interval might better match the full-order model.

In order to investigate this, a study was undertaken to vary the starting point of the training interval. The starting point for the training interval is a snapshot of the flow from our initial transient simulation. We varied which snapshot was used to restart `SPARC` by up to ±2 snapshots. For each case, we ran the full-order model in `SPARC` to collect snapshots of the flow, and then computed a separate POD basis for each case.

Figure 3.9 shows the results when the start of the training interval is shifted forward in time by 1 snapshot from the transient simulation. For this case, the ROM does not seem to exhibit the shift that occurs at the end of the training interval, and so it may more accurately reproduce the behavior of the FOM. Figure 3.10 shows the results when the start of the training interval is shifted even more forward in time. Figures 3.11 and 3.12 complete the study by shifting the starting point backward in time by 1 or 2 snapshots, respectively.

These results appear to demonstrate that the choice of the training interval has a significant impact

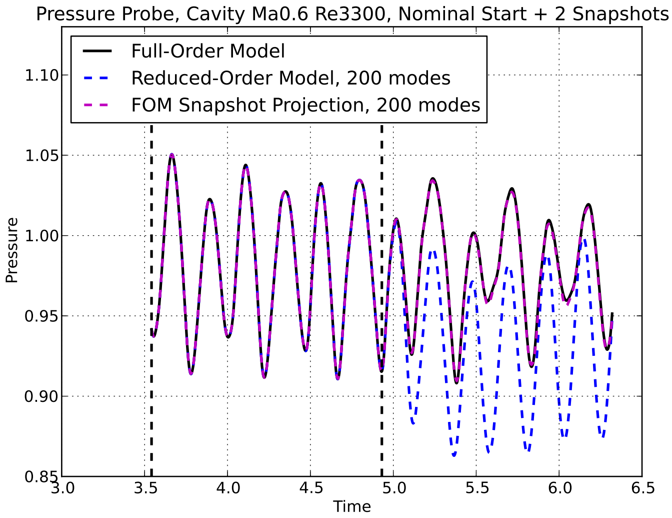


Figure 3.10. The pressure time history for a point midway up the downstream wall of the cavity. The starting point is shifted forward in time by 2 snapshot from the nominal case. Shown are results for the full-order model, reduced-order model using 200 modes, and the full-order model snapshots projected onto 200 modes of the POD basis. The vertical dashed lines indicate the extent of the training data used to create the POD basis.

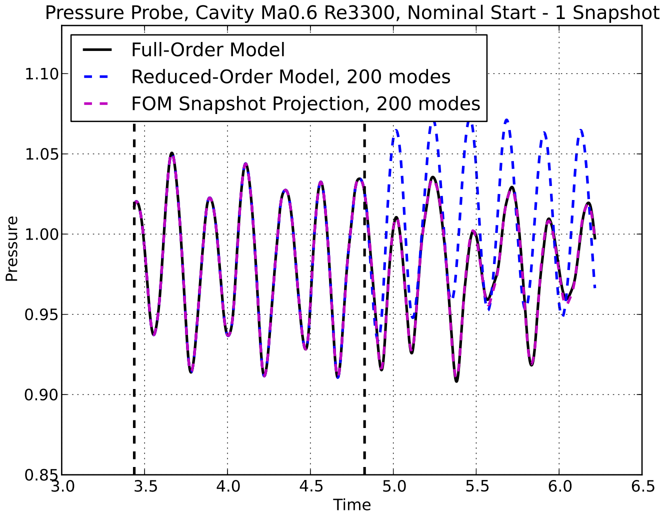


Figure 3.11. The pressure time history for a point midway up the downstream wall of the cavity. The starting point is shifted backward in time by 1 snapshot from the nominal case. Shown are results for the full-order model, reduced-order model using 200 modes, and the full-order model snapshots projected onto 200 modes of the POD basis. The vertical dashed lines indicate the extent of the training data used to create the POD basis.

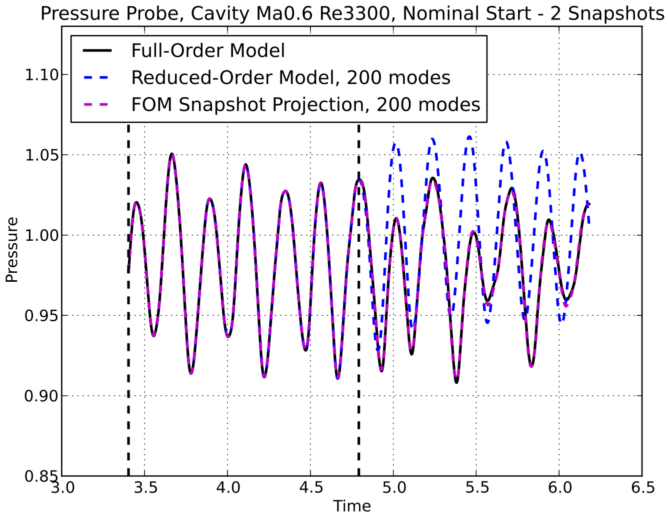


Figure 3.12. The pressure time history for a point midway up the downstream wall of the cavity. The starting point is shifted backward in time by 2 snapshots from the nominal case. Shown are results for the full-order model, reduced-order model using 200 modes, and the full-order model snapshots projected onto 200 modes of the POD basis. The vertical dashed lines indicate the extent of the training data used to create the POD basis.

on the accuracy of the reduced-order models. However, the pressure time histories are only for a single point. It is possible that we are doing better at this point in some cases, but the accuracy at other locations may be worse. Figure 3.13 shows the errors over the entire computational domain for all of these cases. As can be seen, the errors are comparable for all the training intervals both before and after the end of the training interval.

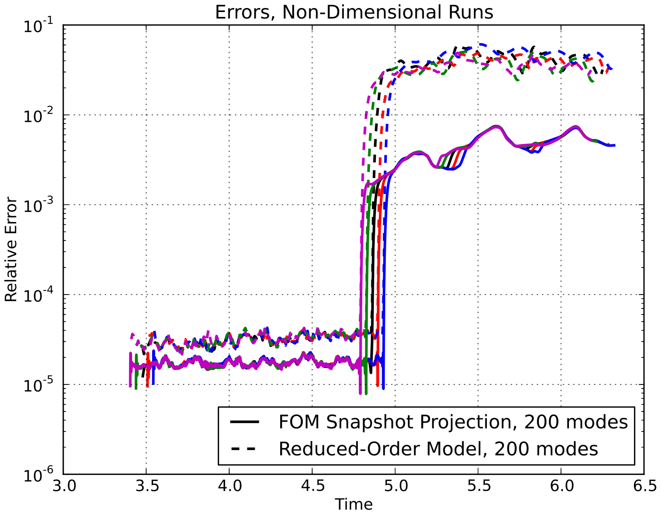


Figure 3.13. The relative errors of the reduced-order model runs compared to projecting the full-order model snapshots onto the POD basis. The solid lines are for the FOM snapshots projected onto the POD basis, and the dashed lines are for the reduced-order models. The different colors represent different training intervals.

3.4.3 Vary length of training window

The length of the training window is expected to have an effect on the accuracy of the reduced-order models. The training window must be long enough so that the snapshots produced capture the important features of the flow. The resulting POD basis should then be able to accurately reproduce the true flow field. However, making the training window too large increases the offline cost of the ROM and may reduce the cost effectiveness of using a ROM. The primary goal of this work is to train a ROM on a short time simulation and then use it to predict further in time. If a simulation of a certain time length is required to generate the appropriate power spectra, then having a short training window maximizes the benefit of using a ROM. It is therefore important to have a training window long enough to capture the important flow features, but not too long as to make the ROM not cost effective.

For the results presented so far, the training interval consists of 2000 time steps using either the dimensional or non-dimensional time steps given in Table 3.1. In previous work during FY16, we primarily used a smaller training interval of 800 time steps. This smaller training interval was deemed insufficient, leading to the 2000 time step interval. However, expanding the training interval further may allow for more accurate reduced-order models.

The following results use a training interval consisting of 4000 time steps. Snapshots of the flow field are collected every time step and used to create a POD basis consisting of 4000 modes. Figure 3.14 shows the pressure time history for reduced-order models using 100 and 200 modes from this new basis. The 100 mode ROM experienced a non-physical result and terminated before a time of 7. The 300 mode ROM ran for the full time that was requested, an additional 4000 time steps beyond the end of the training interval. An untruncated ROM using 4000 modes is not available because of the large computational cost associated with this many modes. Improvements such as hyper-reduction or QR factorization may help reduce the cost and allow for this type of run in the future.

The 300 mode ROM using this larger training interval does not show as dramatic a shift as the ROMs using the 2000 time step interval, as shown in Figure 3.5. However, it was previously seen that shifting the training interval forward or backward in time had a significant impact on the pressure time histories. Figures 3.15, 3.16, 3.17, and 3.18 show the results of shifting the training interval. These plots verify the observation that the pressure time history at a point is affected by the choice of the training interval. However, it is expected that the error of the entire flow field will exhibit similar trends to Figure 3.13 and be insensitive to the training interval start time.

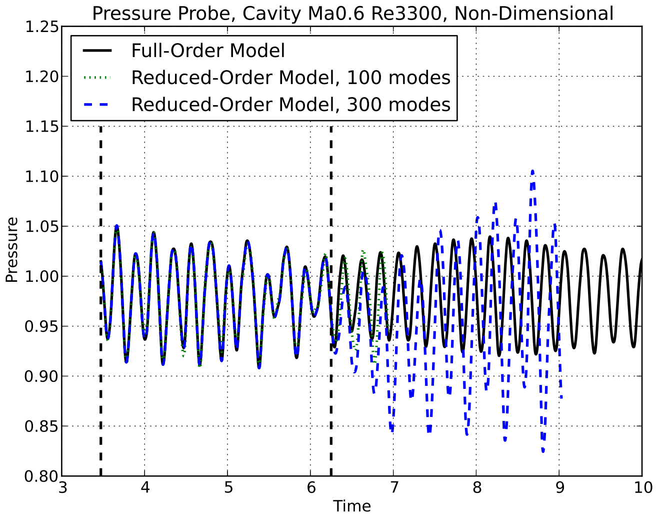


Figure 3.14. The pressure time history for a point midway up the downstream wall of the cavity. Shown are results for the full-order model and reduced-order model non-dimensional runs. The vertical dashed lines indicate the extent of the training data used to create the POD basis used by the reduced-order models.

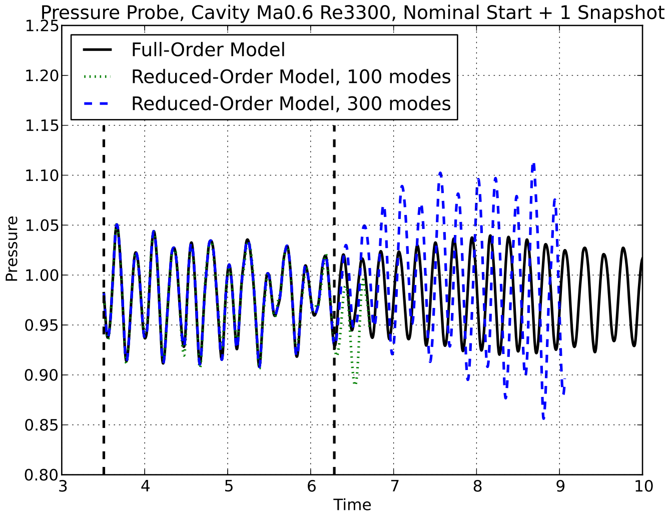


Figure 3.15. The pressure time history for a point midway up the downstream wall of the cavity. The starting point is shifted forward in time by 1 snapshot from the nominal case. Shown are results for the full-order model and reduced-order model runs. The vertical dashed lines indicate the extent of the training data used to create the POD basis.

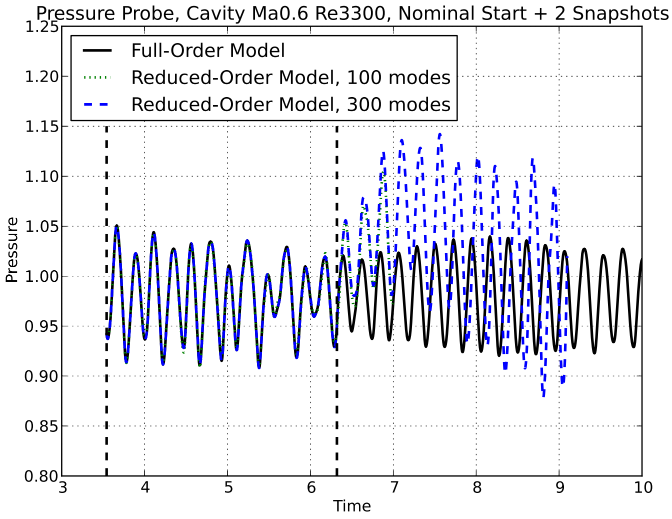


Figure 3.16. The pressure time history for a point midway up the downstream wall of the cavity. The starting point is shifted forward in time by 2 snapshot from the nominal case. Shown are results for the full-order model and reduced-order model runs. The vertical dashed lines indicate the extent of the training data used to create the POD basis.

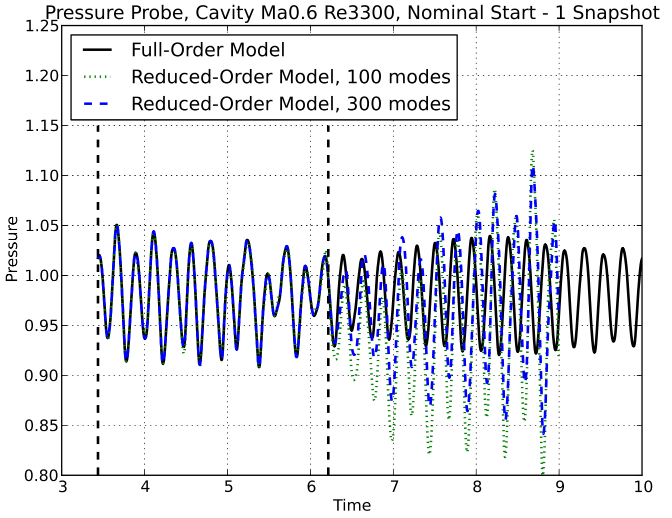


Figure 3.17. The pressure time history for a point midway up the downstream wall of the cavity. The starting point is shifted backward in time by 1 snapshot from the nominal case. Shown are results for the full-order model and reduced-order model runs. The vertical dashed lines indicate the extent of the training data used to create the POD basis.

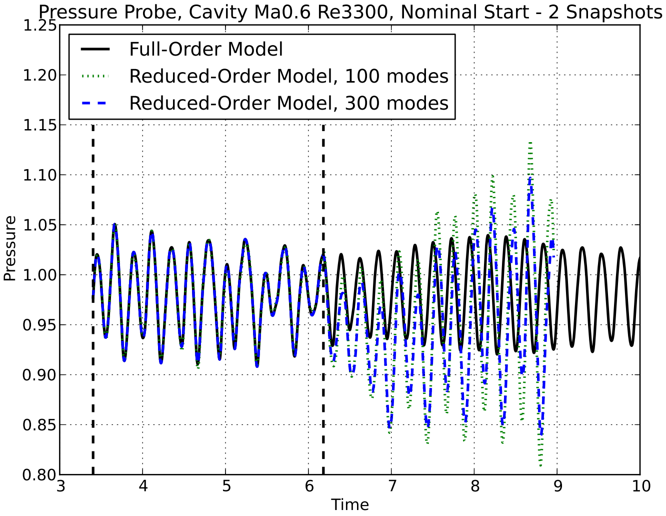


Figure 3.18. The pressure time history for a point midway up the downstream wall of the cavity. The starting point is shifted backward in time by 2 snapshots from the nominal case. Shown are results for the full-order model and reduced-order model runs. The vertical dashed lines indicate the extent of the training data used to create the POD basis.

3.4.4 Project FOM solution increment onto POD basis

In Section 3.4.1, we discussed projecting the snapshots of the flow field created by the FOM onto the POD basis in order to compute projection coefficients. These projection coefficients were then used to construct an approximation to the flow state at each time instance using (3.3). This provides information on the accuracy to which the flow state could be represented using the POD basis. However, it does not take into account the fact that the true FOM flow states may never be reached by the ROM, and this accuracy may never be realized in practice.

In order to determine the upper limit on the ROM accuracy, we need to take the solution increment that is computed by the FOM at each time step, project it onto the POD basis, and then construct an approximated solution increment. If $\Delta \boldsymbol{w}^{(k)}$ is the solution increment computed by the FOM at time step k, then

$$\Delta \tilde{\boldsymbol{w}}^{(k)} = \boldsymbol{\Phi}_M \left(\boldsymbol{\Phi}_M^T \boldsymbol{\Phi}_M \right)^{-1} \boldsymbol{\Phi}_M^T \Delta \boldsymbol{w}^{(k)} \tag{3.4}$$

is the approximated solution increment. This approximated solution increment is then used to update the flow state, producing a flow state that is realizable by an ideal preconditioned LSPG ROM.

Ideal preconditioned LSPG ROMs

The projected solution increment given by (3.4) is equivalent to what would be produced by a preconditioned LSPG ROM using an exact preconditioner. A preconditioned LSPG ROM can be formed by inserting a preconditioner, $\boldsymbol{M}$, into (2.8) to create

$$\delta \hat{\boldsymbol{w}}_{\mathrm{PG}}^{(k)} = \arg \min_{\boldsymbol{y} \in \mathbb{R}^M} \left\| \boldsymbol{M} \left(\boldsymbol{J}^{(k)} \boldsymbol{\Phi}_M \boldsymbol{y} + \boldsymbol{r}^{(k)} \right) \right\|_2^2, \quad k = 1, \ldots, K_{\mathrm{PG}} \,. \tag{3.5}$$

The insertion of a preconditioner has the effect of altering the norm we are minimizing in. The corresponding normal-equations form of (3.5) is

$$\boldsymbol{\Phi}_M^T \boldsymbol{J}^{(k)T} \boldsymbol{M}^T \boldsymbol{M} \boldsymbol{J}^{(k)} \boldsymbol{\Phi}_M \delta \hat{\boldsymbol{w}}_{\mathrm{PG}}^{(k)} = -\boldsymbol{\Phi}_M^T \boldsymbol{J}^{(k)T} \boldsymbol{M}^T \boldsymbol{M} \boldsymbol{r}^{(k)}, \quad k = 1, \ldots, K_{\mathrm{PG}} \,. \tag{3.6}$$

Using an ideal preconditioner

$$\boldsymbol{M} = \left(\boldsymbol{J}^{(k)} \right)^{-1} \tag{3.7}$$

reduces the normal-equations to

$$\boldsymbol{\Phi}_M^T \boldsymbol{\Phi}_M \delta \hat{\boldsymbol{w}}_{\mathrm{PG}}^{(k)} = -\boldsymbol{\Phi}_M^T \left(\boldsymbol{J}^{(k)} \right)^{-1} \boldsymbol{r}^{(k)}, \quad k = 1, \ldots, K_{\mathrm{PG}} \,. \tag{3.8}$$

Substituting in the true FOM solution increment,

$$\Delta \boldsymbol{w}^{(k)} = \left(\boldsymbol{J}^{(k)} \right)^{-1} \boldsymbol{r}^{(k)} \,, \tag{3.9}$$

produces

$$\boldsymbol{\Phi}_M^T \boldsymbol{\Phi}_M \delta \hat{\boldsymbol{w}}_{\mathrm{PG}}^{(k)} = -\boldsymbol{\Phi}_M^T \Delta \boldsymbol{w}^{(k)}, \quad k = 1, \ldots, K_{\mathrm{PG}} \,. \tag{3.10}$$

The right hand side is the projection of the true FOM solution increment. Solving for $\delta \hat{\boldsymbol{w}}_{\mathrm{PG}}^{(k)}$ and constructing the approximate solution increment produces (3.4).

Alternative preconditioner definitions

In `SPARC`, the conservative variable solution increment, $\Delta \boldsymbol{w}^{(k)}$ is not computed directly. An intermediate increment is first computed, which is then converted to the conservative variable solution increment by dividing by the cell volume. In other words, `SPARC` solves

$$\boldsymbol{A}(\boldsymbol{w})\Delta\boldsymbol{p} = -\boldsymbol{b}(\boldsymbol{w}) \ , \tag{3.11}$$

where $\boldsymbol{A}(\boldsymbol{w})$ is an approximate flux Jacobian matrix without the inclusion of the cell volumes and $\boldsymbol{b}(\boldsymbol{w})$ is the flux vector residual. The intermediate solution increment, $\Delta\boldsymbol{p}$, is then converted to the conservative variable solution increment by dividing by the cell volume. In other words,

$$\Delta\boldsymbol{w} = \boldsymbol{V}^{-1}\Delta\boldsymbol{p} \ , \tag{3.12}$$

where $\boldsymbol{V}$ is a diagonal matrix with the cell volumes on the diagonal.

The normal-equation form for a standard LSPG ROM in `SPARC` is therefore

$$[\boldsymbol{V}\boldsymbol{\Phi}_M]^T[\boldsymbol{A}(\tilde{\boldsymbol{w}})]^T[\boldsymbol{A}(\tilde{\boldsymbol{w}})][\boldsymbol{V}\boldsymbol{\Phi}_M]\Delta\hat{\boldsymbol{w}} = -[\boldsymbol{V}\boldsymbol{\Phi}_M]^T[\boldsymbol{A}(\tilde{\boldsymbol{w}})]^T\boldsymbol{b}(\tilde{\boldsymbol{w}}) \ . \tag{3.13}$$

The volume matrix is grouped with the POD basis because both are constant and their product can be precomputed. Introducing a preconditioner gives

$$[\boldsymbol{V}\boldsymbol{\Phi}_M]^T[\boldsymbol{A}(\tilde{\boldsymbol{w}})]^T\boldsymbol{M}^T\boldsymbol{M}[\boldsymbol{A}(\tilde{\boldsymbol{w}})][\boldsymbol{V}\boldsymbol{\Phi}_M]\Delta\hat{\boldsymbol{w}} = -[\boldsymbol{V}\boldsymbol{\Phi}_M]^T[\boldsymbol{A}(\tilde{\boldsymbol{w}})]^T\boldsymbol{M}^T\boldsymbol{M}\boldsymbol{b}(\tilde{\boldsymbol{w}}) \ . \tag{3.14}$$

The ideal preconditioner given in (3.7) therefore corresponds to

$$\boldsymbol{M} = [\boldsymbol{A}(\tilde{\boldsymbol{w}})\boldsymbol{V}]^{-1} \ . \tag{3.15}$$

This use of this preconditioner will lead to an optimal ℓ^2 projection of the solution increment $\Delta\boldsymbol{w}$ onto the basis $\boldsymbol{\Phi}_M$. Inserting this preconditioner into (3.14) will produce (3.4), or equivalently

$$\Delta\tilde{\boldsymbol{w}} = \arg\min_{\mathbf{x}} \|\boldsymbol{\Phi}_M\mathbf{x} - \Delta\boldsymbol{w}\|_2 \ . \tag{3.16}$$

Alternative forms for the preconditioner can also be defined, such as

$$\boldsymbol{M} = [\boldsymbol{A}(\tilde{\boldsymbol{w}})]^{-1} \tag{3.17}$$

or

$$\boldsymbol{M} = [\boldsymbol{A}(\tilde{\boldsymbol{w}})\boldsymbol{V}^{\frac{1}{2}}]^{-1} \ . \tag{3.18}$$

The first alternative leads to an approximated solution increment of the form

$$\Delta\tilde{\boldsymbol{p}}^{(k)} = [\boldsymbol{V}\boldsymbol{\Phi}_M]\left([\boldsymbol{V}\boldsymbol{\Phi}_M]^T[\boldsymbol{V}\boldsymbol{\Phi}_M]\right)^{-1}[\boldsymbol{V}\boldsymbol{\Phi}_M]^T\Delta\boldsymbol{p}^{(k)} \ . \tag{3.19}$$

The second alternative leads to an approximated solution increment of the form

$$\Delta\tilde{\boldsymbol{p}}^{(k)} = [\boldsymbol{V}\boldsymbol{\Phi}_M]\left(\boldsymbol{\Phi}_M^T\boldsymbol{V}\boldsymbol{\Phi}_M\right)^{-1}\boldsymbol{\Phi}_M^T\Delta\boldsymbol{p}^{(k)} \ . \tag{3.20}$$

Note that these two alternatives operate on the intermediate solution increment to produce an approximated intermediate solution increment. The approximated intermediate solution increment would then be scaled by the cell volumes to produce the approximated conservative-variable solution increment $\Delta\tilde{\boldsymbol{w}}$.

The first alternative then becomes

$$\Delta\tilde{\boldsymbol{w}}^{(k)} = \boldsymbol{\Phi}_M \left(\boldsymbol{\Phi}_M^T \boldsymbol{V}^2 \boldsymbol{\Phi}_M\right)^{-1} \boldsymbol{\Phi}_M^T \boldsymbol{V}^2 \Delta\boldsymbol{w}^{(k)} \,, \tag{3.21}$$

and the second alternative becomes

$$\Delta\tilde{\boldsymbol{w}}^{(k)} = \boldsymbol{\Phi}_M \left(\boldsymbol{\Phi}_M^T \boldsymbol{V} \boldsymbol{\Phi}_M\right)^{-1} \boldsymbol{\Phi}_M^T \boldsymbol{V} \Delta\boldsymbol{w}^{(k)} \,. \tag{3.22}$$

The second alternative is equivalent to

$$\Delta\tilde{\boldsymbol{w}} = \arg\min_{\mathbf{x}} \|\boldsymbol{\Phi}_M \mathbf{x} - \Delta\boldsymbol{w}\|_{L^2(\Omega)} \,, \tag{3.23}$$

where

$$\|\mathbf{x}\|_{L^2(\Omega)} = \mathbf{x}^T \boldsymbol{V} \mathbf{x} \,. \tag{3.24}$$

The first alternative uses the norm

$$\|\mathbf{x}\|_{L^2(\Omega)} = \mathbf{x}^T \boldsymbol{V}^2 \mathbf{x} \,. \tag{3.25}$$

Projected solution increment results

Figure 3.19 shows a comparison of the three projected solution increment implementations discussed above. For comparison, the full-order model and standard LSPG reduced-order model are also shown. These runs used an untruncated POD basis of 2000 modes in order to provide the best results possible for each approach.

There are some interesting observations that can be made from this plot. The standard projected solution increment given in (3.4) seems to exhibit the correct frequency behavior, but the amplitudes of the oscillations are generally larger than the true full-order model signal. The first alternative implementation, given by (3.19), seems to exhibit similar behavior to the non-preconditioned LSPG ROM. The second alternative implementation, given by (3.20), seems to have nearly correct amplitudes for the oscillations, but the frequency or phase is a little off, especially towards the end of the run.

As was observed previously, the precise start of the training interval affects the pressure time histories for the reduced-order models. Figures 3.20, 3.21, 3.22, and 3.23 show the effect of shifting the training interval on the projected solution increment runs. While shifting the training interval does have an impact on the results, the relative behavior observed previously for the three projected solution increment implementations remains the same.

The results presented so far have been for a fairly limited simulation time of twice the snapshot collection time. This has been useful to focus attention on what happens at the end of the training

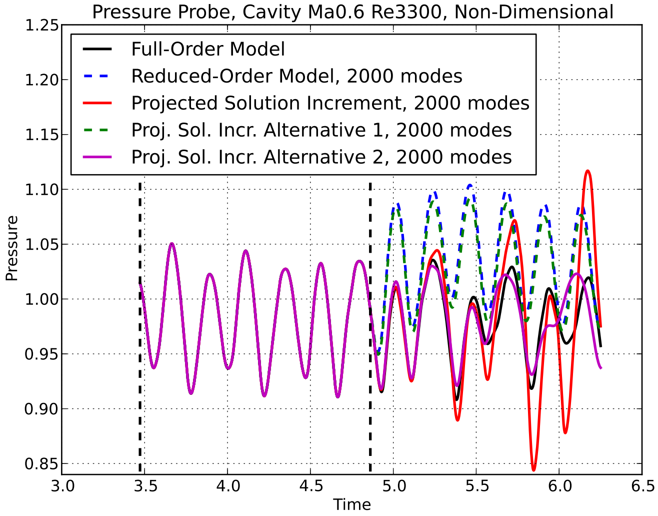


Figure 3.19. The pressure time history for a point midway up the downstream wall of the cavity. Shown are results for the full-order model, reduced-order model, and projected solution increment runs using an untruncated POD basis of 2000 modes. The vertical dashed lines indicate the extent of the training data used to create the POD basis.

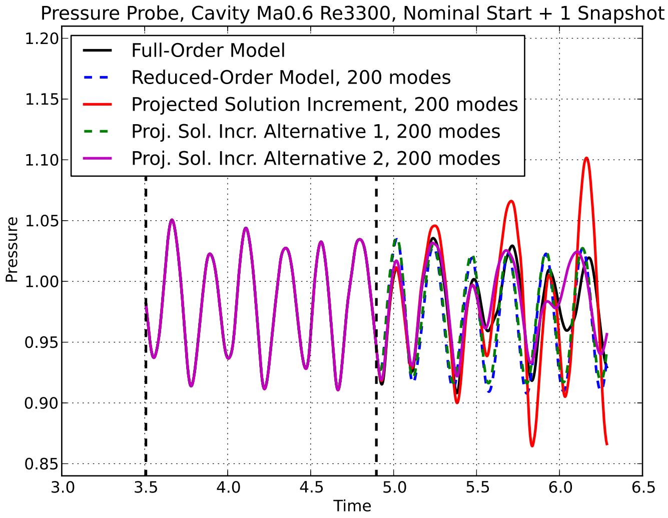


Figure 3.20. The pressure time history for a point midway up the downstream wall of the cavity. The starting point is shifted forward in time by 1 snapshot from the nominal case. Shown are results for the full-order model, reduced-order model, and projected solution increment runs. The vertical dashed lines indicate the extent of the training data used to create the POD basis.

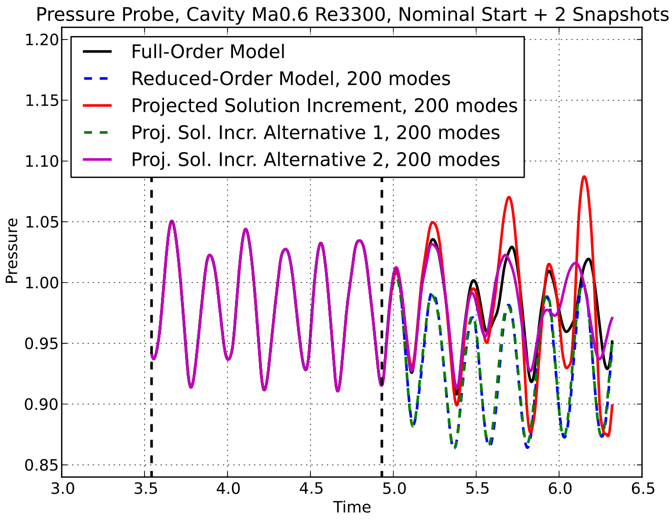


Figure 3.21. The pressure time history for a point midway up the downstream wall of the cavity. The starting point is shifted forward in time by 2 snapshot from the nominal case. Shown are results for the full-order model, reduced-order model, and projected solution increment runs. The vertical dashed lines indicate the extent of the training data used to create the POD basis.

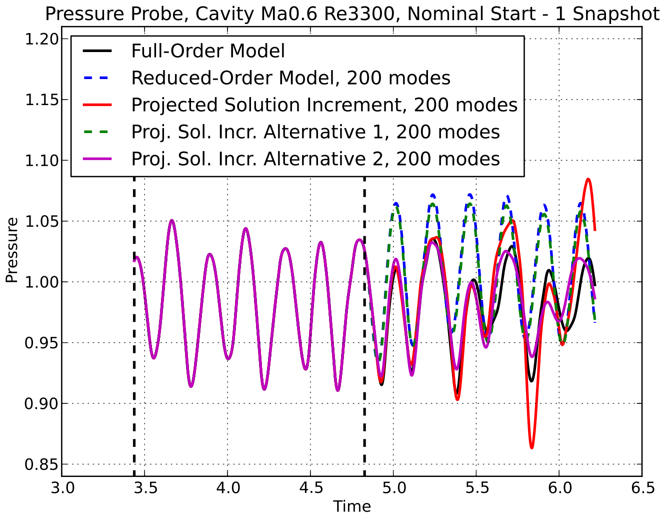


Figure 3.22. The pressure time history for a point midway up the downstream wall of the cavity. The starting point is shifted backward in time by 1 snapshot from the nominal case. Shown are results for the full-order model, reduced-order model, and projected solution increment runs. The vertical dashed lines indicate the extent of the training data used to create the POD basis.

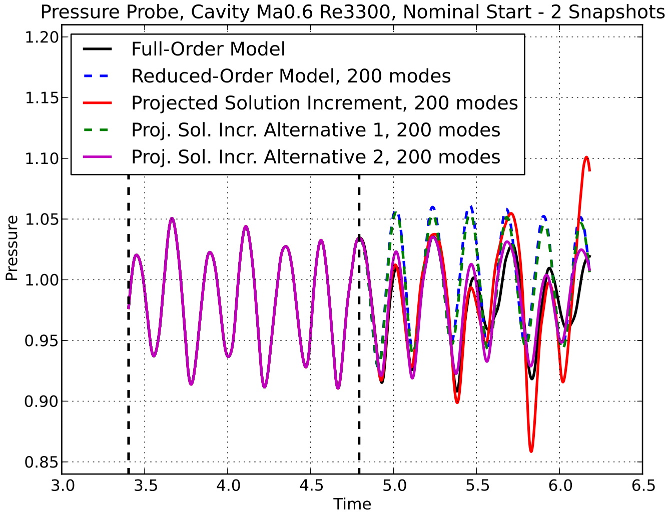


Figure 3.23. The pressure time history for a point midway up the downstream wall of the cavity. The starting point is shifted backward in time by 2 snapshots from the nominal case. Shown are results for the full-order model, reduced-order model, and projected solution increment runs. The vertical dashed lines indicate the extent of the training data used to create the POD basis.

interval. However, we are interested in using ROMs for long time predictions, so the following results will consider a simulation time of 40,000 time steps. Figure 3.24 shows the pressure time histories for the full-order model and the three projected solution increment implementations. For the standard projected solution increment given in (3.4), shown in the top plot, the pressure fluctuations have a larger amplitude that those for the full-order model. For the first alternative implementation, given by (3.19) and shown in the middle plot, the mean of pressure fluctuations jumps to a higher value at the training interval boundary. The amplitude of the fluctuations then begins to decay. It should be pointed out that the data for this case is incomplete due to a computer issue and it was not deemed worth it to rerun the case. For the second alternative implementation, given by (3.20) and shown in the bottom plot, the amplitudes of the fluctuations appear on par with the full-order model but there is discrepancy in frequency or phase.

Ultimately, we are interested in whether the ROMs produce acceptable power spectra for the pressure, not whether we exactly replicate the pressure time history. Figure 3.25 shows the power spectral density of the pressure for the full-order model and the three projected solution increment implementations. For the standard projected solution increment, given in (3.4) and shown in the top plot, we seem to accurately capture the frequency and amplitude for the first three peaks. However, in general there is a shift of the PSD upwards. This is to be expected since the area under the PSD is the rms of the pressure fluctuations, and the pressure time history exhibited larger amplitude fluctuations. For the first alternative implementation, given by (3.19) and shown in the middle plot, we seem to capture the second peak fairly well but otherwise it does not seem to be very accurate. For the second alternative implementation, given by (3.20) and shown in the bottom plot, the first

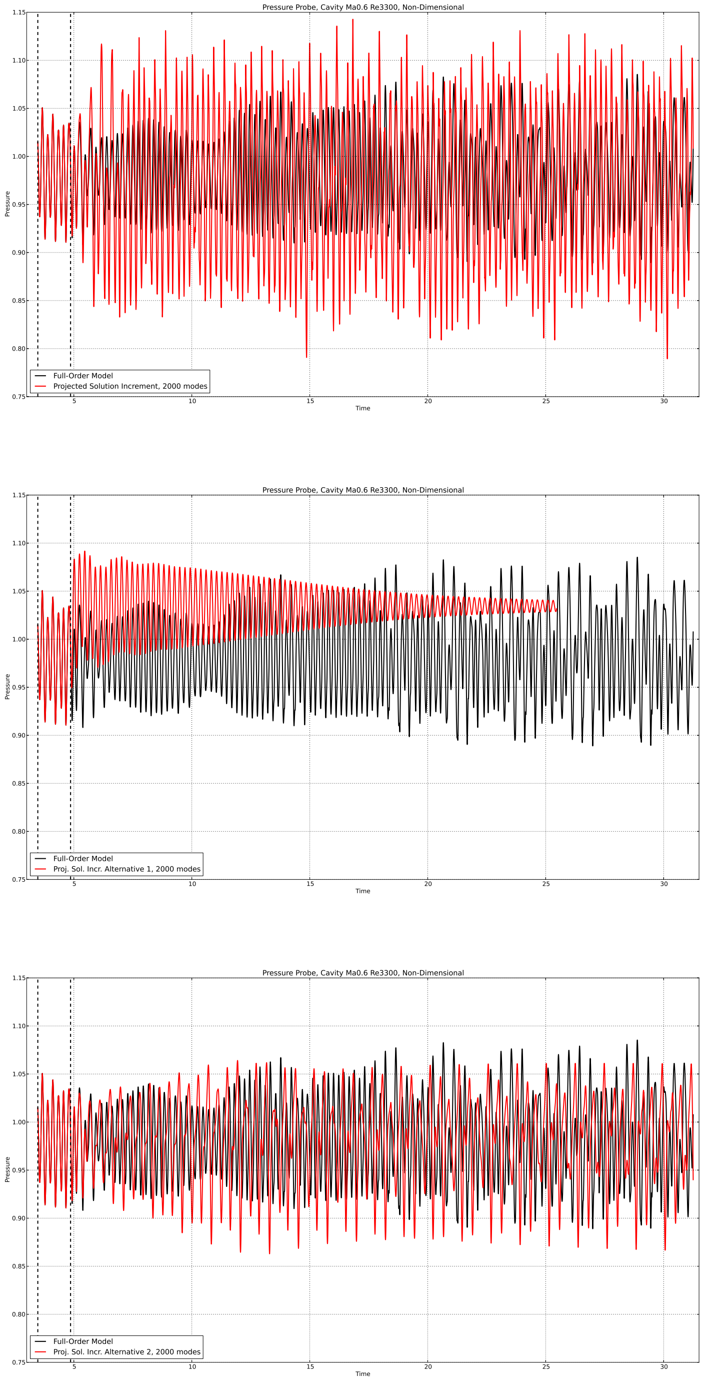


Figure 3.24. The pressure time history for a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the three projected solution increment implementations using an untruncated POD basis of 2000 modes. The vertical dashed lines indicate the extent of the training data used to create the POD basis.

few peaks seem to match fairly well although there is some noise in the PSD.

The next set of results are for a training interval of 4000 time steps, as discussed in Section 3.4.3. The simulations are still run for a total of 40,000 time steps. This means that the training interval is 10% of the total simulation time, in contrast to 5% for the previous results. The resultant ROMs should be more accurate, but they have lost some of their cost benefit. Figure 3.26 shows the pressure time histories for the full-order model and the three projected solution increment implementations. It should be noted that the projected solution increment simulations are incomplete. For the middle plot, this is because `SPARC` encountered an error, possibly a non-physical result (e.g., a negative pressure or temperature) and terminated early. For the top and bottom plots, the runs terminated early due to a computer issue and have not been rerun yet. For the standard projected solution increment given in (3.4), shown in the top plot, the pressure fluctuations exhibit behavior, such as a beating pattern, that is qualitatively similar to the full-order model. For this case the amplitudes of the fluctuations are smaller than the FOM, in contrast to the shorter training interval results. For the first alternative implementation, given by (3.19) and shown in the middle plot, the simulation terminates too early to make any reasonable comparisons. However, the fact that the simulation quit early indicates that it may have exhibited non-physical behavior. For the second alternative implementation, given by (3.20) and shown in the bottom plot, the fluctuations grow in amplitude after the end of the training interval and then reach a constant amplitude oscillation. This behavior is quite different than the observed behavior on the shorter training interval.

Again, we are interested in whether the ROMs produce acceptable power spectra for the pressure, not whether we exactly replicate the pressure time history. Figure 3.27 shows the power spectral density of the pressure for the full-order model and the three projected solution increment implementations. For the standard projected solution increment, given in (3.4) and shown in the top plot, we seem to accurately capture the frequency and amplitude of the first few peaks, with the exception of the fourth and sixth peaks. For the first alternative implementation, given by (3.19) and shown in the middle plot, the results are deceptively good because the simulation did not get much beyond the training interval. For the second alternative implementation, given by (3.20) and shown in the bottom plot, the second peak is captured fairly well but based on the time history we do not expect to match other frequencies.

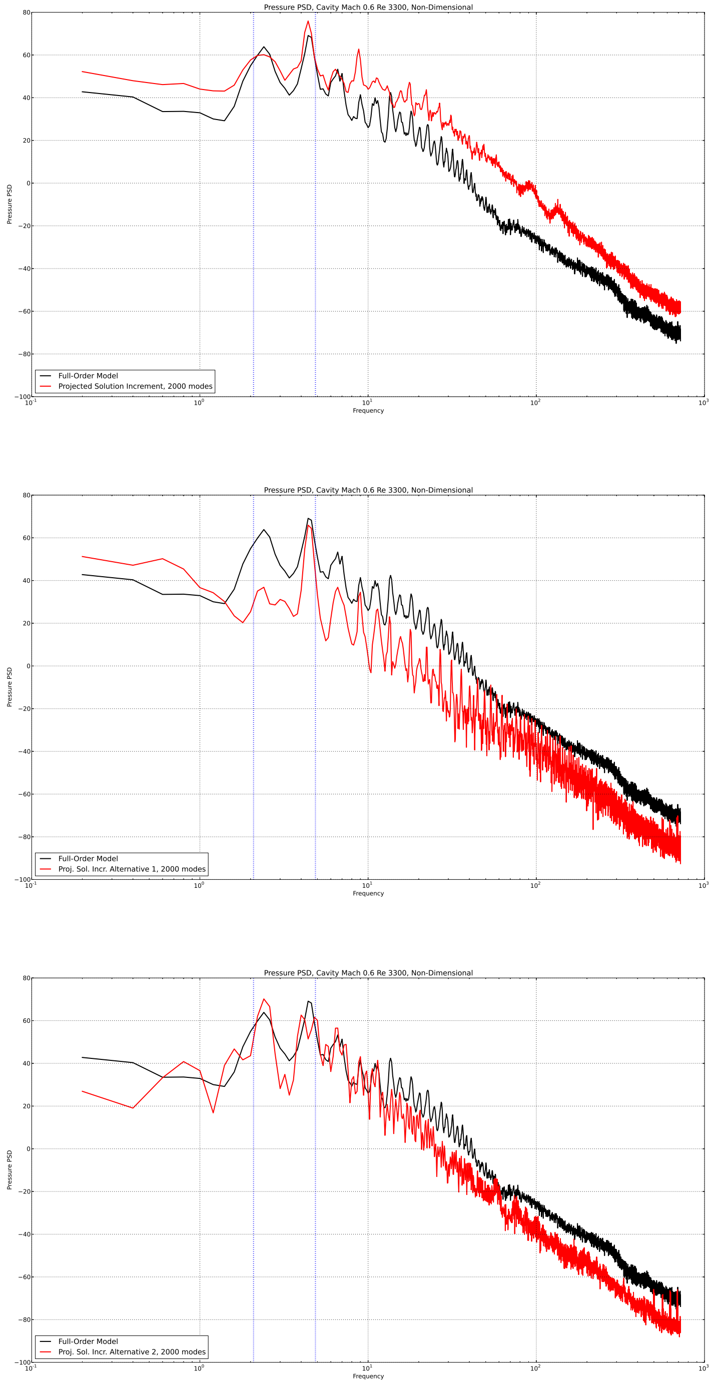


Figure 3.25. The power spectral density of the pressure at a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the three projected solution increment implementations using an untruncated POD basis of 2000 modes. The vertical dotted lines indicate the first and second Rossiter tones.

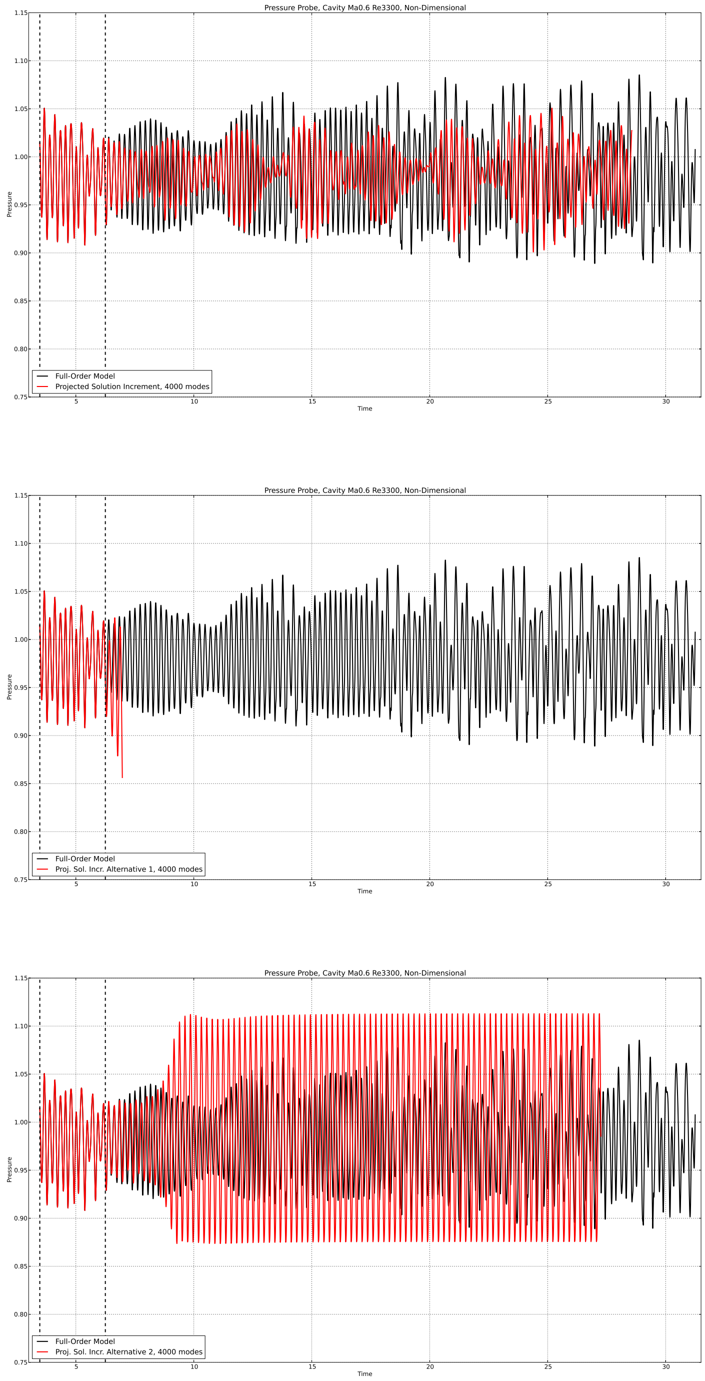


Figure 3.26. The pressure time history for a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the three projected solution increment implementations using an untruncated POD basis of 4000 modes. The vertical dashed lines indicate the extent of the training data used to create the POD basis.

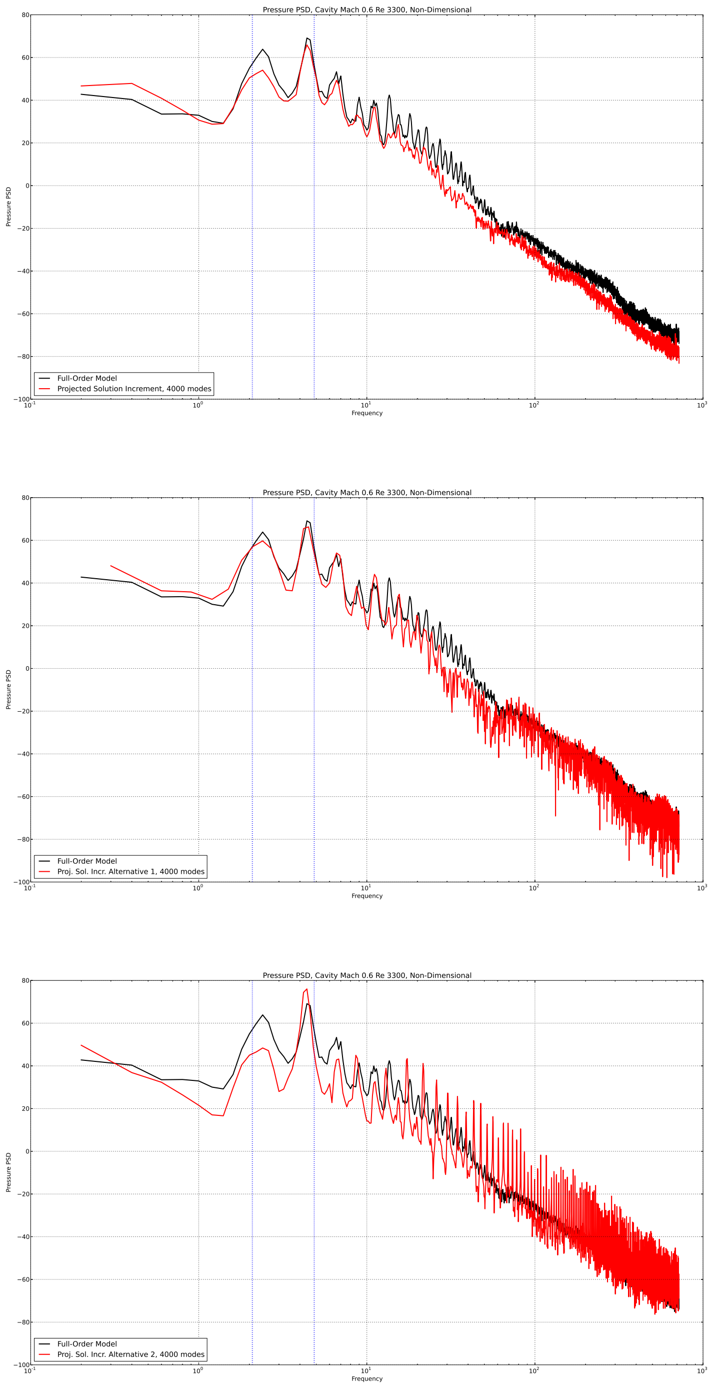


Figure 3.27. The power spectral density of the pressure at a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the three projected solution increment implementations using an untruncated POD basis of 4000 modes. The vertical dotted lines indicate the first and second Rossiter tones.

3.4.5 Time-step refinement

The time step is an important parameter affecting the accuracy of LSPG ROMs. In [7], error bounds were derived that showed a dependence on the time step. It was found that there is an optimal time step which minimizes the error of the LSPG ROMs, and that error increases if the time step is too small. This was verified computationally for a cavity flow using the LSPG ROM implementation in AERO-F. The optimal time step was found to be larger than that used for the full-order model.

The results presented so far have all been created using the same time step as that used for the full-order model. In this section, the effect of varying the time step will be assessed. The results presented here will focus on the impact of varying the time step for runs using the standard projected solution increment given in (3.4), and the second alternative implementation given by (3.20), as these produced the best results using the nominal time step. For this assessment, the time step will be set to 2×, 5×, 10×, 20×, 50×, and 100× the nominal value given in Table 3.1.

The results in this section are incomplete. Some cases are still running at the time of this report. On the following plots, if the data does not extend to the end of the simulation time then it is because the case had not finished running when the plot was generated. Figure 3.28 shows the pressure time history and the PSD that result from running the standard projected solution increment implementation with 2× the nominal time step. It is difficult to tell from the time history if this is an improvement over Figure 3.24. The high frequency part of the PSD is lower than that in Figure 3.25 and may be more accurate overall, however the new run is not finished so the PSD uses only a limited time history.

Figure 3.29 shows the pressure time history and the PSD that result from running the standard projected solution increment implementation with 5× the nominal time step. The pressure time history shows that the simulation becomes periodic with constant amplitude. Figure 3.30 shows the pressure time history and the PSD that result from running the standard projected solution increment implementation with 10× the nominal time step. Figure 3.31 shows the pressure time history and the PSD that result from running the standard projected solution increment implementation with 20× the nominal time step. The pressure time histories for both the 10× and 20× runs seem to exhibit behavior that is more similar to the FOM than the projected solution increment using the nominal time step or 2× nominal. The resulting PSDs show that the low frequency peaks match the FOM fairly well, while the high frequency content is below that of the FOM.

Figure 3.32 shows the pressure time history and the PSD that result from running the standard projected solution increment implementation with 50× the nominal time step. Figure 3.33 shows the pressure time history and the PSD that result from running the standard projected solution increment implementation with 100× the nominal time step. These results do not look very good, which suggests that too large of a time step was used.

Figures 3.34, 3.35, 3.36, 3.37, 3.38, and 3.39 show the effect of varying the time step when using the second alternative projected solution increment implementation. These results do not look as good as those for the standard projected solution increment presented earlier. This is despite the results for the nominal time step looking better than those of the standard implementation.

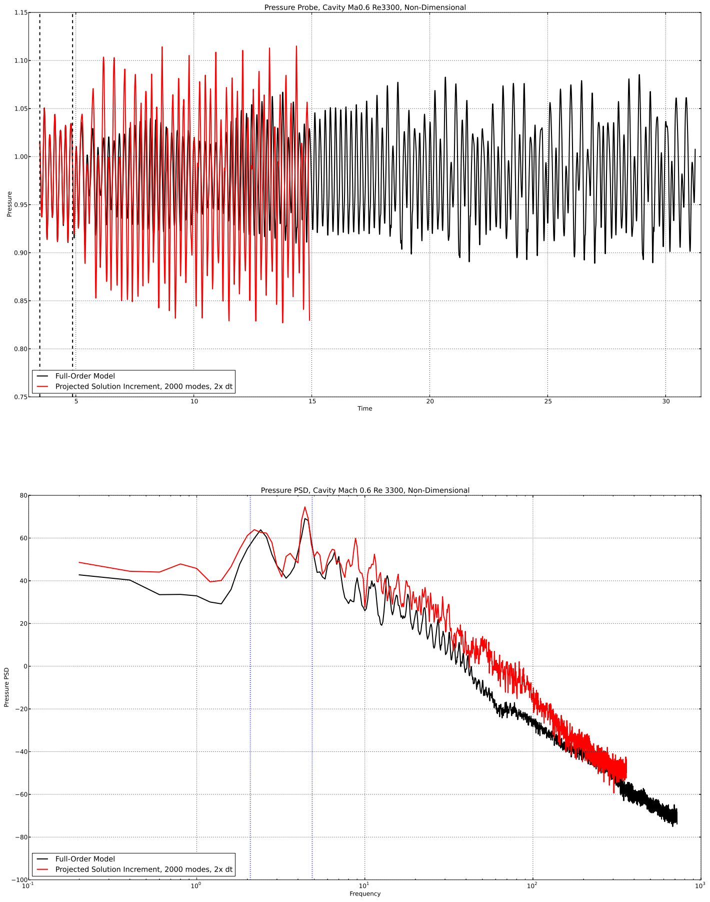


Figure 3.28. The pressure time history and resulting power spectral at a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the standard projected solution increment implementation using an untruncated POD basis of 2000 modes. The projected solution increment case is run using 2× the nominal time step. The vertical dashed lines on the time history plot indicate the extent of the training data used to create the POD basis. The vertical dotted lines on the PSD plot indicate the first and second Rossiter tones.

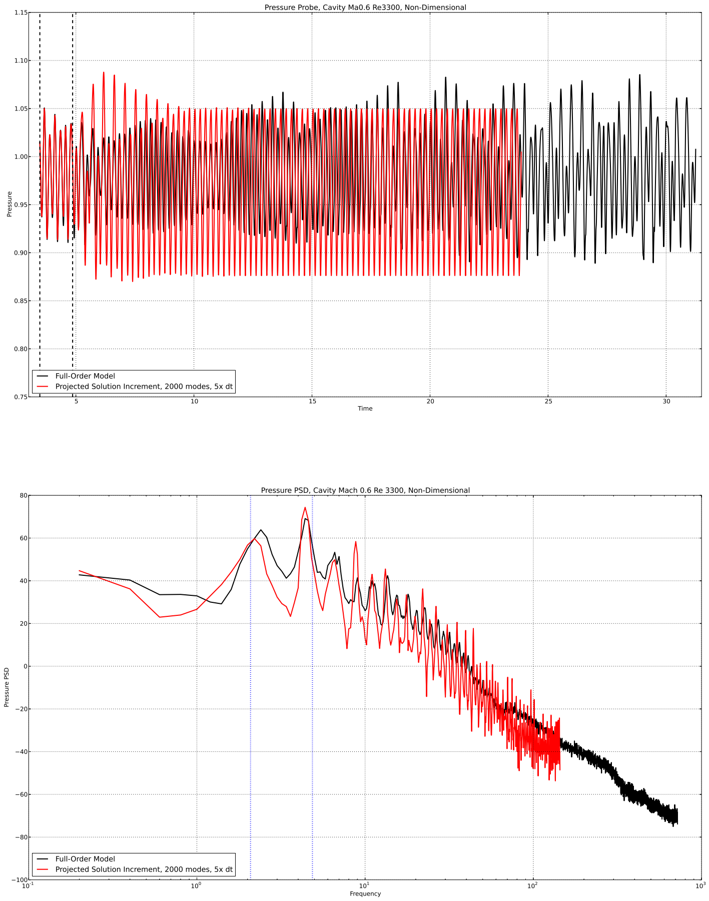


Figure 3.29. The pressure time history and resulting power spectral at a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the standard projected solution increment implementation using an untruncated POD basis of 2000 modes. The projected solution increment case is run using 5× the nominal time step. The vertical dashed lines on the time history plot indicate the extent of the training data used to create the POD basis. The vertical dotted lines on the PSD plot indicate the first and second Rossiter tones.

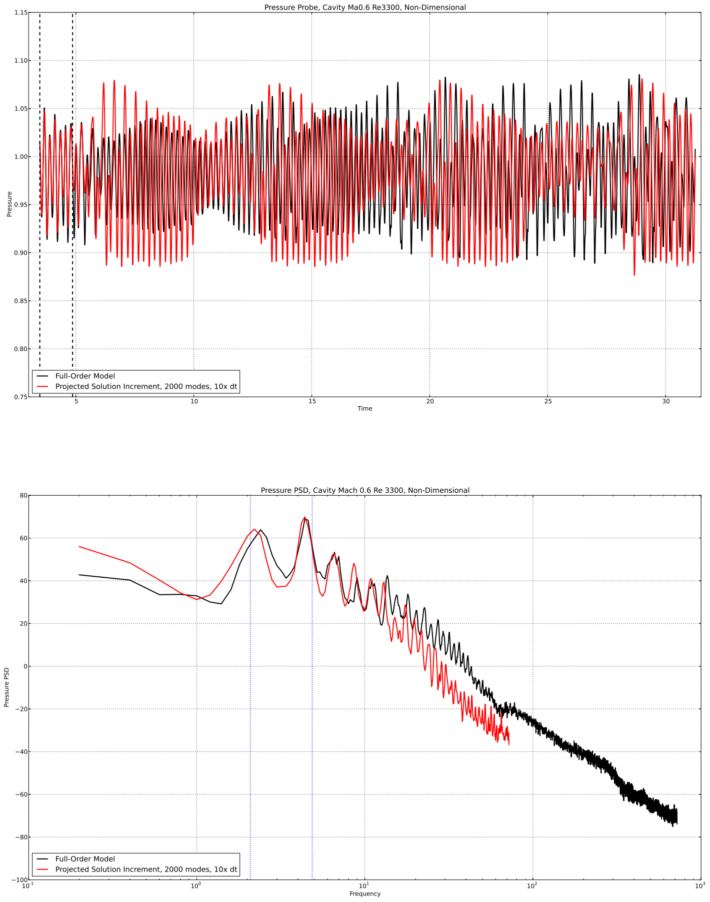


Figure 3.30. The pressure time history and resulting power spectral at a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the standard projected solution increment implementation using an untruncated POD basis of 2000 modes. The projected solution increment case is run using 10× the nominal time step. The vertical dashed lines on the time history plot indicate the extent of the training data used to create the POD basis. The vertical dotted lines on the PSD plot indicate the first and second Rossiter tones.

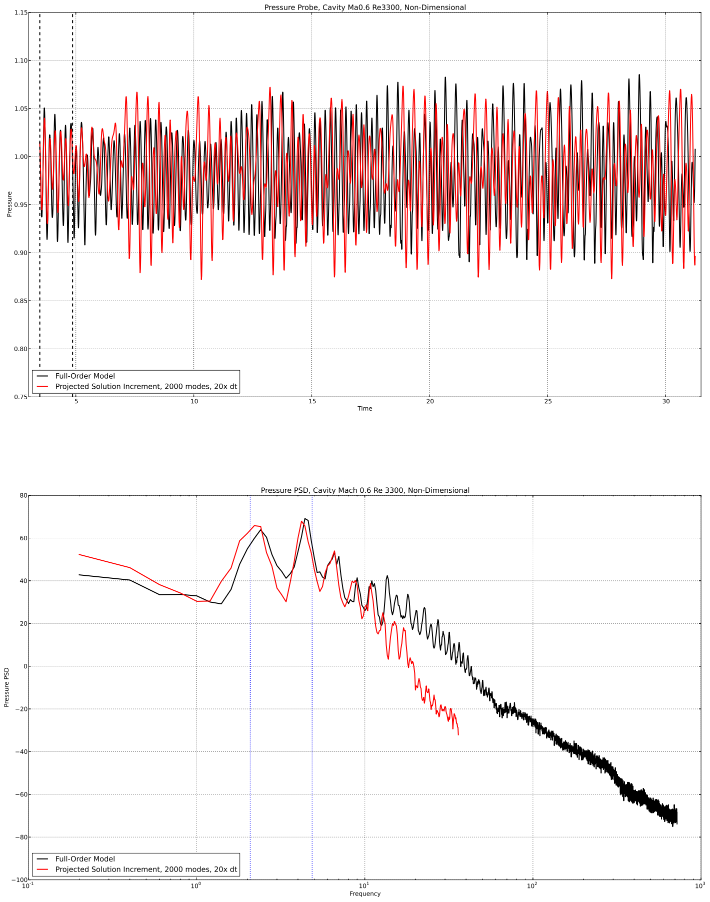


Figure 3.31. The pressure time history and resulting power spectral at a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the standard projected solution increment implementation using an untruncated POD basis of 2000 modes. The projected solution increment case is run using 20× the nominal time step. The vertical dashed lines on the time history plot indicate the extent of the training data used to create the POD basis. The vertical dotted lines on the PSD plot indicate the first and second Rossiter tones.

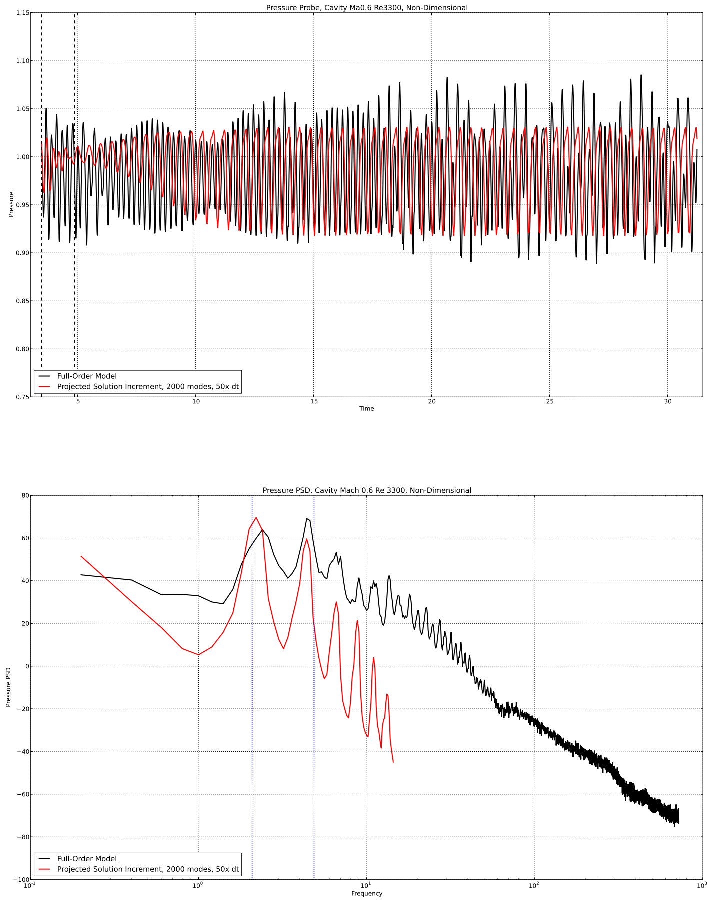


Figure 3.32. The pressure time history and resulting power spectral at a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the standard projected solution increment implementation using an untruncated POD basis of 2000 modes. The projected solution increment case is run using 50× the nominal time step. The vertical dashed lines on the time history plot indicate the extent of the training data used to create the POD basis. The vertical dotted lines on the PSD plot indicate the first and second Rossiter tones.

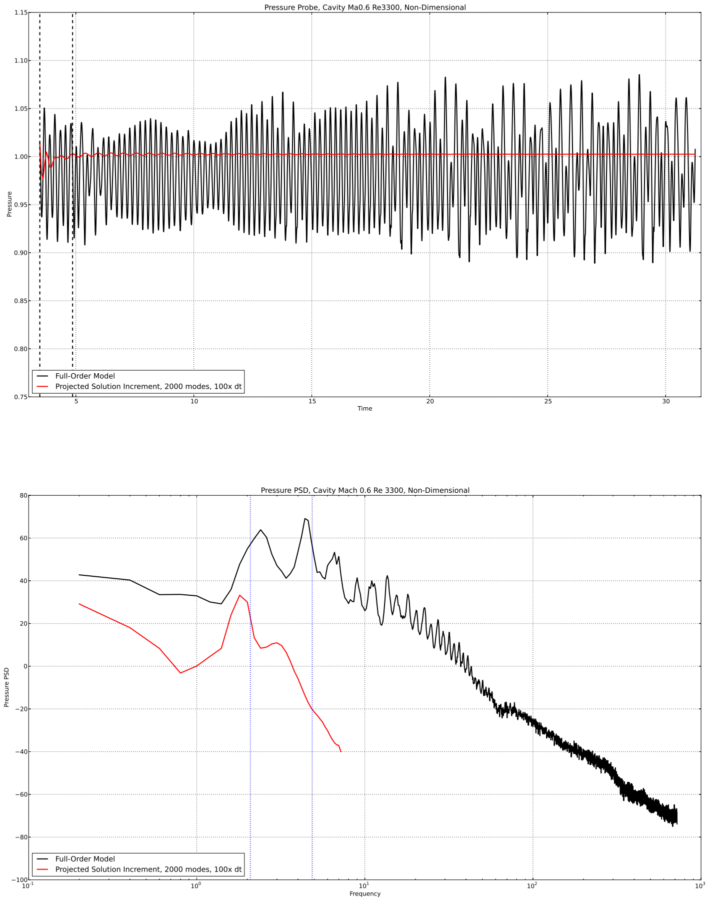


Figure 3.33. The pressure time history and resulting power spectral at a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the second alternative projected solution increment implementation using an untruncated POD basis of 2000 modes. The projected solution increment case is run using 100× the nominal time step. The vertical dashed lines on the time history plot indicate the extent of the training data used to create the POD basis. The vertical dotted lines on the PSD plot indicate the first and second Rossiter tones.

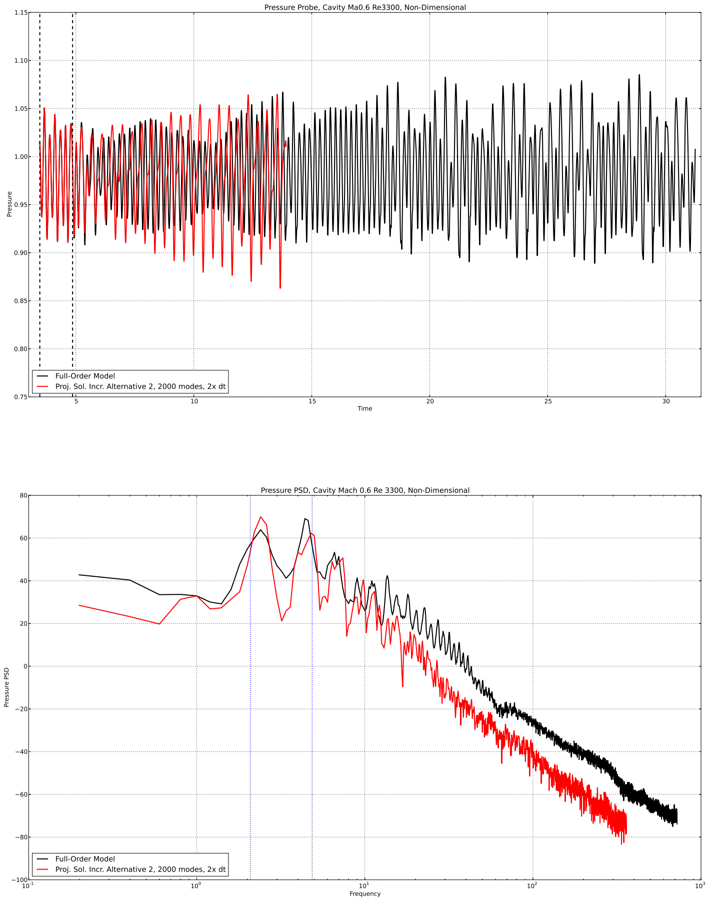


Figure 3.34. The pressure time history and resulting power spectral at a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the second alternative projected solution increment implementation using an untruncated POD basis of 2000 modes. The projected solution increment case is run using 2× the nominal time step. The vertical dashed lines on the time history plot indicate the extent of the training data used to create the POD basis. The vertical dotted lines on the PSD plot indicate the first and second Rossiter tones.

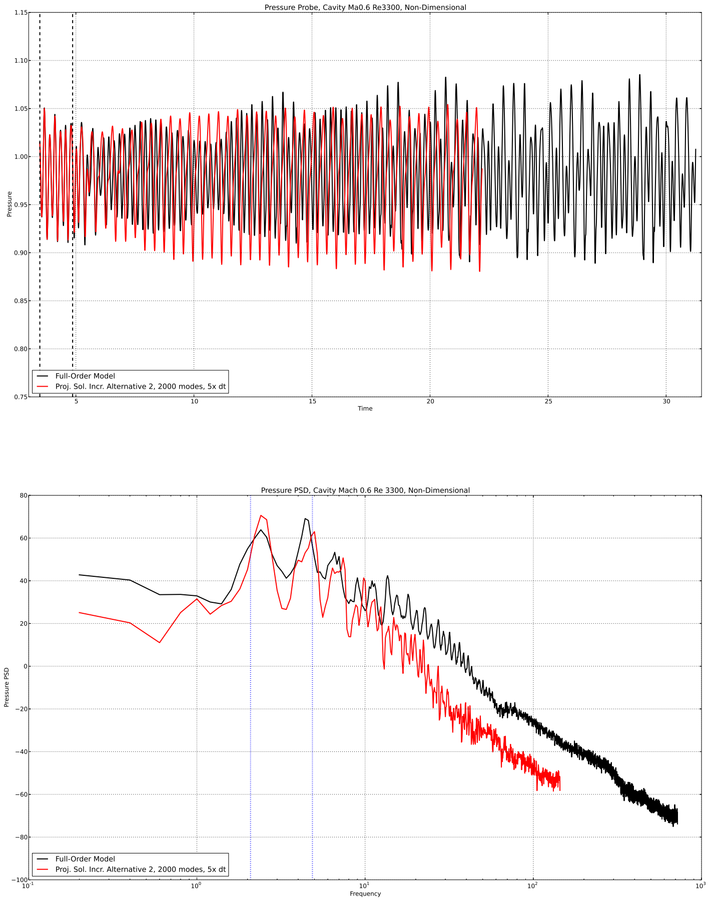


Figure 3.35. The pressure time history and resulting power spectral at a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the second alternative projected solution increment implementation using an untruncated POD basis of 2000 modes. The projected solution increment case is run using 5× the nominal time step. The vertical dashed lines on the time history plot indicate the extent of the training data used to create the POD basis. The vertical dotted lines on the PSD plot indicate the first and second Rossiter tones.

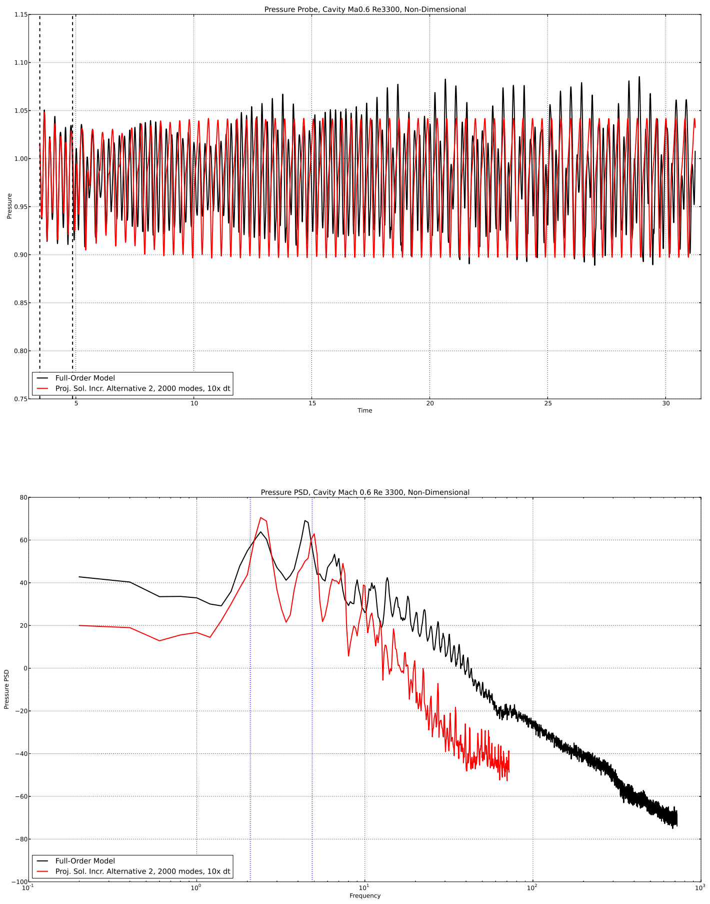


Figure 3.36. The pressure time history and resulting power spectral at a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the second alternative projected solution increment implementation using an untruncated POD basis of 2000 modes. The projected solution increment case is run using 10× the nominal time step. The vertical dashed lines on the time history plot indicate the extent of the training data used to create the POD basis. The vertical dotted lines on the PSD plot indicate the first and second Rossiter tones.

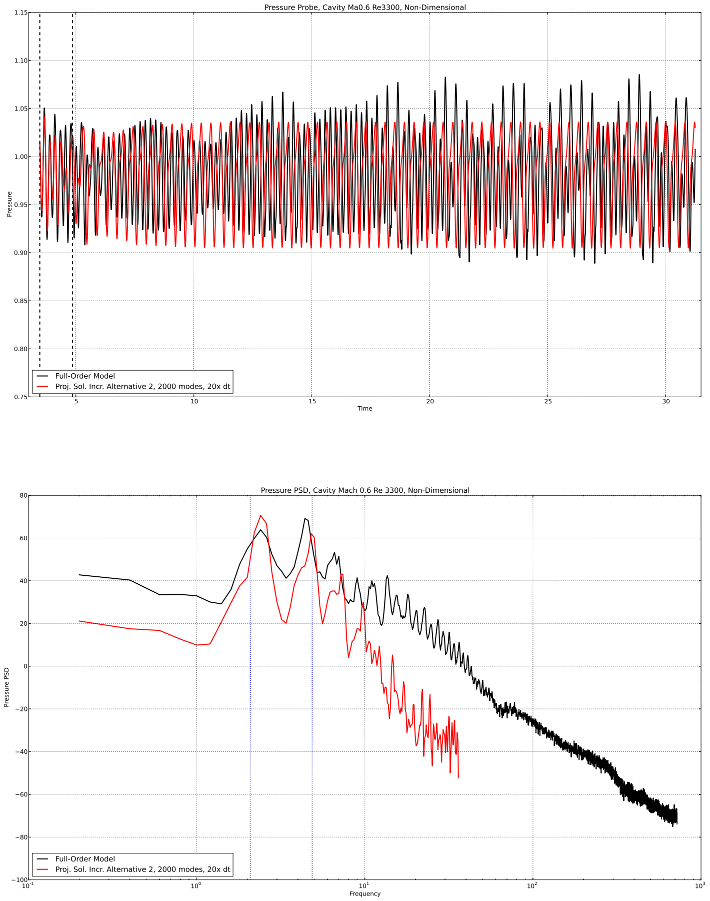


Figure 3.37. The pressure time history and resulting power spectral at a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the second alternative projected solution increment implementation using an untruncated POD basis of 2000 modes. The projected solution increment case is run using 20× the nominal time step. The vertical dashed lines on the time history plot indicate the extent of the training data used to create the POD basis. The vertical dotted lines on the PSD plot indicate the first and second Rossiter tones.

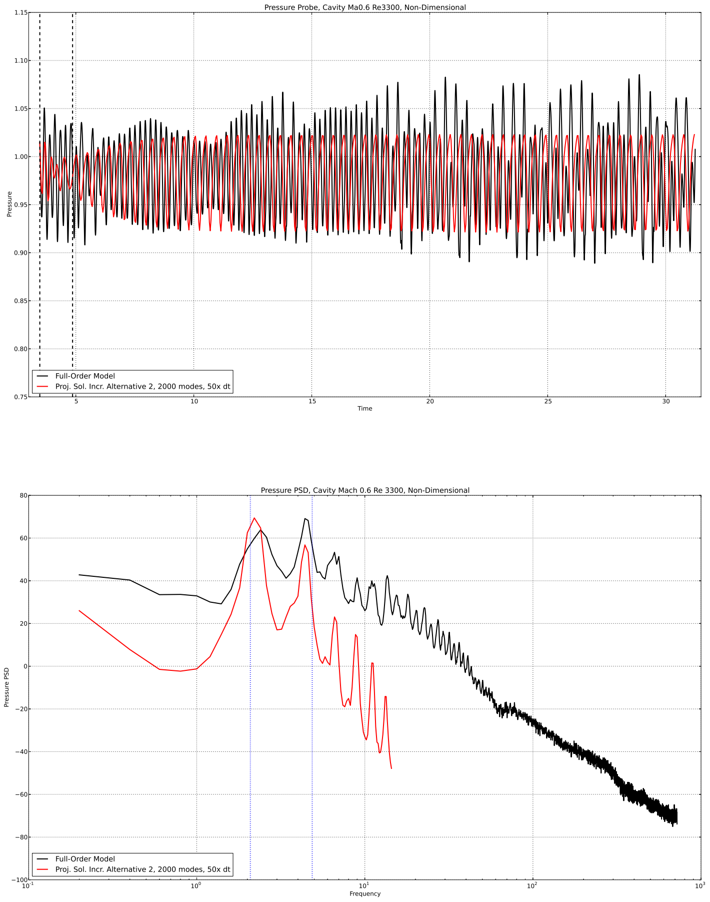


Figure 3.38. The pressure time history and resulting power spectral at a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the second alternative projected solution increment implementation using an untruncated POD basis of 2000 modes. The projected solution increment case is run using 50× the nominal time step. The vertical dashed lines on the time history plot indicate the extent of the training data used to create the POD basis. The vertical dotted lines on the PSD plot indicate the first and second Rossiter tones.

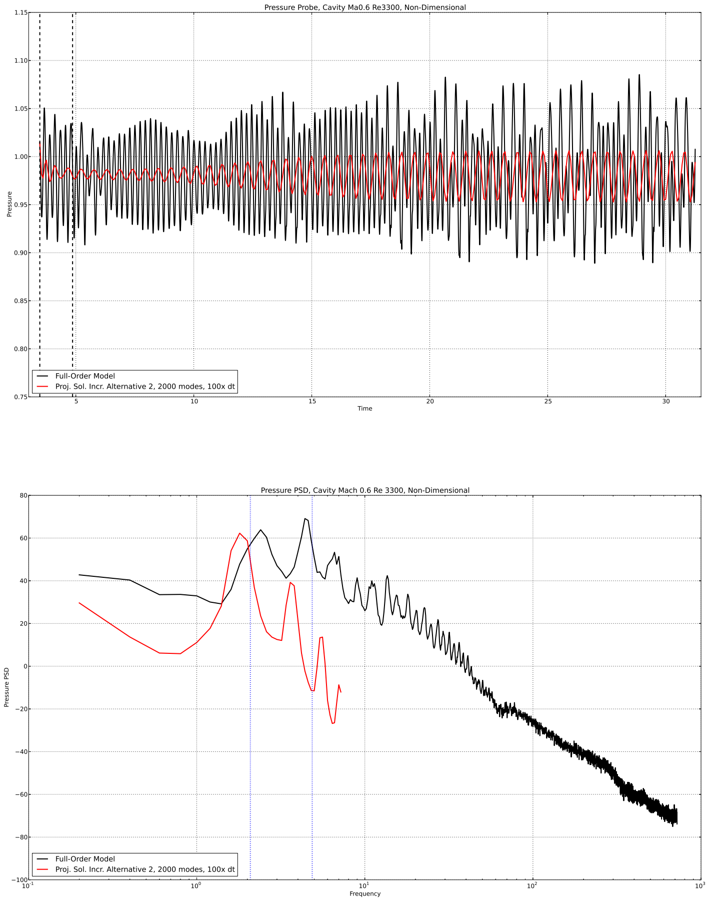


Figure 3.39. The pressure time history and resulting power spectral at a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the second alternative projected solution increment implementation using an untruncated POD basis of 2000 modes. The projected solution increment case is run using 100× the nominal time step. The vertical dashed lines on the time history plot indicate the extent of the training data used to create the POD basis. The vertical dotted lines on the PSD plot indicate the first and second Rossiter tones.

So far, the results in this section have been for the 2000 time step training interval. Figure 3.40 shows the pressure time history and PSD when the standard projected solution increment using an untruncated basis created from the 4000 time step training interval is run with 2× the nominal time step. It is difficult to say whether the pressure time history is better than that in Figure 3.26. It appears that running with the larger time step slightly lowers the high frequency part of the PSD compared to Figure 3.27. Data for runs using other time steps is currently unavailable. It is therefore too early to try to make a conclusion about the effect of the time step on these runs with a larger basis.

A clear conclusion about the effect of varying the time step cannot be made at this time. It should be pointed out that varying the time step results in the ROM not matching the FOM during the training window. So it seems that varying the time step degrades the accuracy during the training window, but may improve the accuracy after the end of the training interval. Further experiments are underway to more fully explore the effect of varying the time step.

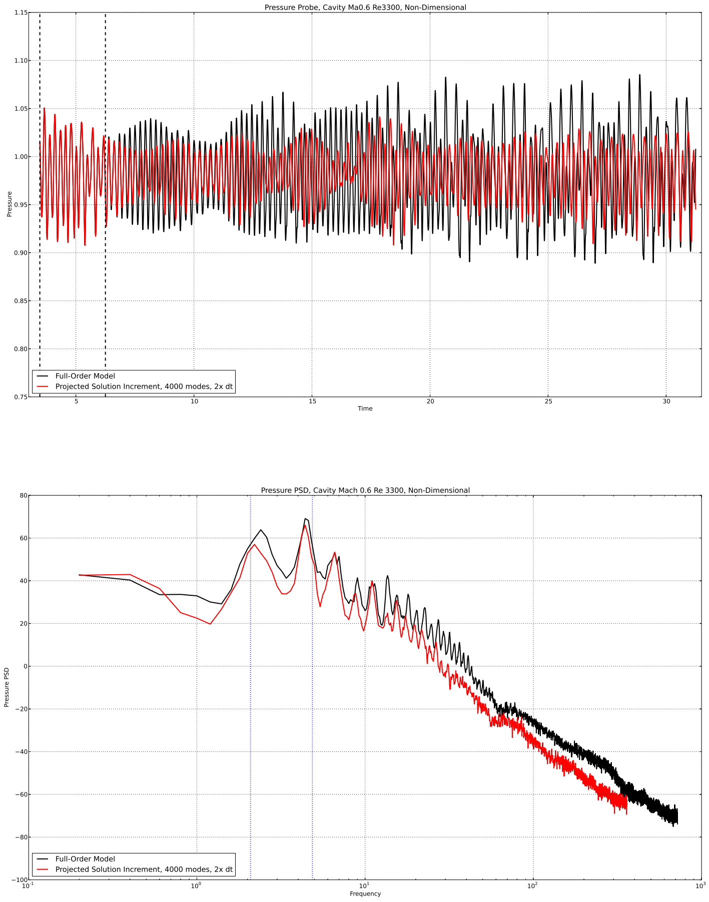


Figure 3.40. The pressure time history and resulting power spectral at a point midway up the downstream wall of the cavity. Shown are results for the full-order model and the standard projected solution increment implementation using an untruncated POD basis of 4000 modes. The projected solution increment case is run using 2× the nominal time step. The vertical dashed lines on the time history plot indicate the extent of the training data used to create the POD basis. The vertical dotted lines on the PSD plot indicate the first and second Rossiter tones.

3.5 Summary and recommendations

Described in this chapter are the results of several `SPARC` LSPG/ROM accuracy assessment studies, which led to recommendations towards improving the predictive capabilities of these models for long-time simulations. The key take-aways from this study are summarized here.

- *Non-dimensionalization.* Running simulations non-dimensionally is believed to be beneficial, as it tends to give rise to systems that are better scaled and hence easier to solve. For some problems, e.g., an inviscid pressure pulse test case, non-dimensionalization drastically improved ROM performance. However, for the viscous cavity the benefit is less clear.

- *Choice of training interval.* We found that the choice of training interval can have a significant impact on the ROM accuracy. Increasing the length of the training window provides more snapshots and more modes and should produce a more accurate ROM, but reduces the cost effectiveness of the ROM. Shifting the training window forward or backward in time has an effect on the local pressure time histories, but may not impact the overall accuracy of the entire flow field.

- *ROM preconditioners.* The ideal preconditioner for the LSPG ROMs is equivalent to projecting the full-order model solution increment onto the basis. This provides an upper bound on how accurate the ROMs can be. The implementation of actual preconditioners that have the effect of using an ideal preconditioner but not the cost in the ROM implementation requires further development and debugging.

- *Choice of time step.* It is unclear if we observe the benefit of increasing the time step reported in [7]. It appears that increasing the time step can improve the accuracy of the long time behavior, but this comes at the expense of decreased accuracy in the training interval. Further experiments are required.

Chapter 4

Implementation and verification of high-order methods in SPARC

An FY17 effort secondary to the development and numerical study of LSPG/POD ROMs in `SPARC` involved the creation of a code verification methodology and workflow, towards enabling the implementation and verification of higher-order discretization methods (e.g., the Rai scheme [35]) in `SPARC`. Although seemingly irrelevant to `SPARC` MOR development, higher-order FOM discretizations in `SPARC` are necessary for 3D cavity ROM simulations using this code base at a later point in time. In this chapter, we summarize the low as well as higher order finite volume schemes currently available in `SPARC`, describe our verification workflow and methodology and present the results of a mesh convergence study in the context of an inviscid pulse test case.

4.1 Summary of flux schemes in `SPARC`

There are two standard flux schemes in `SPARC`: Roe [38] and Steger-Warming [42]. These schemes can be run either first or second order using MUSCL reconstruction to estimate the state at the face of the cell. Recently, low-dissipation Subbareddy-Candler [43] schemes have also been implemented. These low-dissipation schemes can be run either second or fourth order. The implementation of the Rai scheme [35], a fifth-order upwind-biased scheme, is currently underway.

We restrict our attention to examining spatial (vs. temporal) convergence. Although various time-integration schemes are available within `SPARC`, for all the spatial convergence studies described here, the low-order implicit BDF1 scheme was employed.

4.2 Description of verification methodology

In order to verify the spatial accuracy of the flux schemes, we construct a series of uniformly refined meshes. For each flux scheme, the error between the solution on the coarser meshes and the finest mesh is then computed. An exact, analytic solution could also be used if available, but for this work we focused on using the solution on the finest mesh as the true solution.

`SPARC` is a cell-centered finite-volume code, so the computed flow solution is stored at the center of each cell. Starting from a coarse mesh, we create uniformly refined meshes using a refinement factor of 3. In other words, for a 2D mesh, each cell is divided into 9 cells, in a 3×3 pattern. Following this approach, the solution is always available at the spatial location defined by the center of the cells on the coarse mesh. This allows the error between the solutions on different meshes to be computed directly without any interpolation or averaging.

The total error is a discrete l_2 norm of the difference between two flow solutions at all the locations defined by the coarsest mesh under consideration. This error is then divided by the discrete l_2 norm of the solution on the finest mesh in order to produce a relative error, as in (3.2).

4.3 Mesh convergence study: inviscid pulse test case

The inviscid pulse test case, described in detail in [45], is used to verify the spatial accuracy of the flux implementations. This test case consists of an initial Gaussian pressure pulse in a square domain bounded by solid walls which is allowed to evolve in time. A series of meshes was constructed for two related mesh convergence studies. For each study, the mesh was refined by a factor of 3. The preliminary study analyzed the relative error and convergence behavior of a series of four meshes over a time period of 34 seconds. For these runs, the second-order Roe flux scheme was used. Mesh sizes began at 20×20 cells and were refined until 520×540 cells. In this case, the solution on the 520×540 mesh was used as the true solution when computing the error in the coarser meshes.

Figure 4.1 shows the relative error as a function of time. Throughout the steady state period of the run, the error between the 20×20 and 60×60 mesh decreases by a factor of 6. The error between the 60×60 and 180×180 mesh decreases by a factor of 9. In the more comprehensive study described later in this section, this $9\times$ decrease in error for the finer mesh results that use the second-order Roe flux scheme is described in detail.

Figure 4.2 shows a traditional convergence rate plot, where the error at the end of the run is plotted as a function of mesh size. The convergence rate can be computed using the errors between consecutive meshes

$$r_i = \log(E_{i+1}/E_i)/\log(h_{i+1}/h_i)\ , \tag{4.1}$$

where E is the error and h is the mesh size. The reader can observe by inspecting Figure 4.2 that the convergence rate is close to 2, the theoretical convergence rate for the tested second-order Roe flux scheme. Figure 4.3 shows the convergence rate between the refined meshes as a function of time. The blue and red lines in this figure correspond to the slopes of the right and left segment of the line in Figure 4.2, respectively. As the mesh is refined, the asymptotic convergence rate of 2 is achieved.

The comprehensive study included a larger set of mesh refinement sizes and analyzed the full series of flux schemes. The coarsest mesh was 3×3 cells. This mesh was uniformly refined by a factor of 3 to produce a 9×9 cell mesh, where the solution was available at the same locations as the

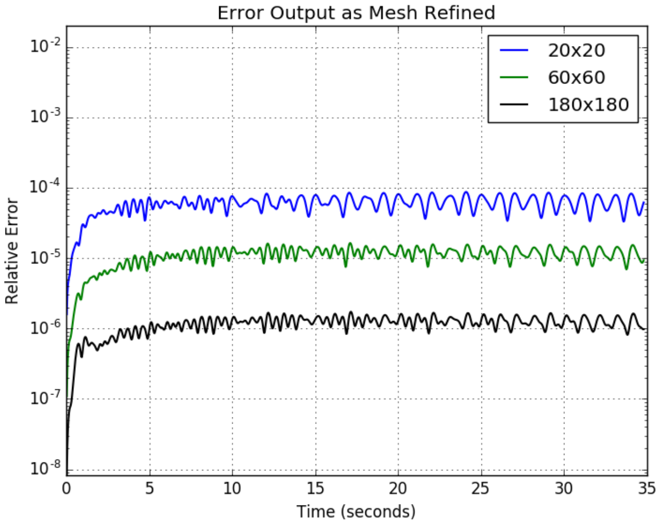


Figure 4.1. The computed relative error between the solution on each mesh and the solution on the finest mesh for the inviscid pulse test case. The results shown are for the second-order Roe flux scheme in `SPARC`.

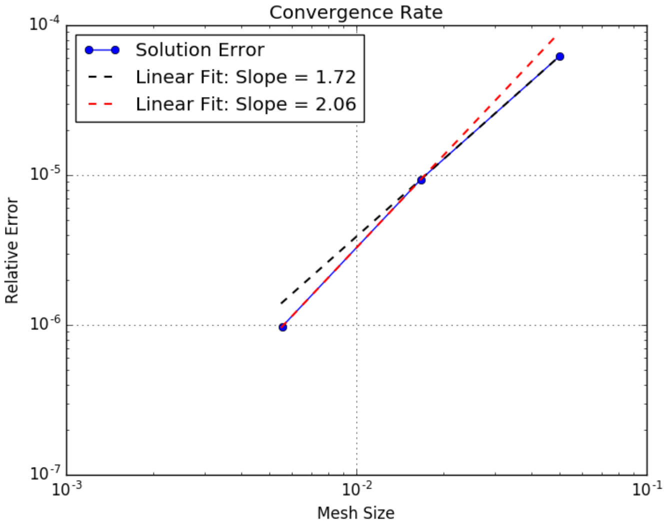


Figure 4.2. The computed relative error as a function of mesh size for the inviscid pulse test case at 34 seconds. The results shown are for the second-order Roe flux scheme in `SPARC`. The convergence rate is close to 2, as expected from theory.

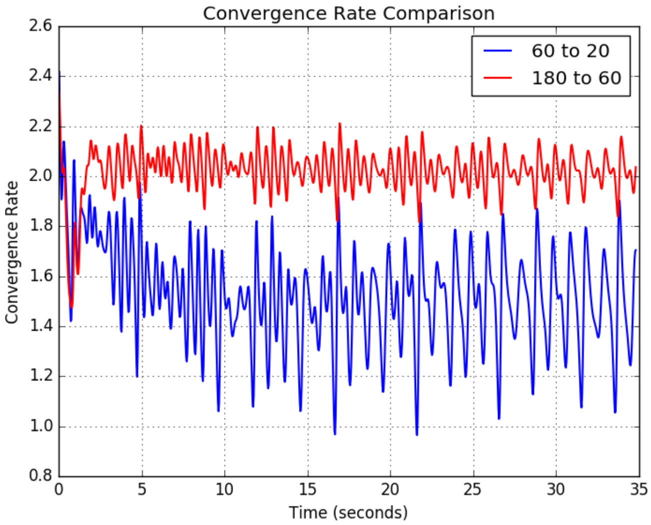


Figure 4.3. The computed convergence rate between mesh sizes as a function of time for the inviscid pulse test case. The results shown are for the second-order Roe flux scheme in `SPARC`. The blue and red lines correspond to the slopes of the right and left segment of the line in Figure 4.2, respectively. As the mesh is refined, the asymptotic convergence rate of 2 is achieved.

3×3 cell mesh. Additional refined meshes were constructed including a 2187×2187 cell mesh, which was the finest mesh constructed. The solution on the 2187×2187 mesh was used as the true solution when computing the errors in the solution on the other meshes. Solution results were computed over a time history of 3 seconds.

Figure 4.4 shows the relative error as a function of time for the first-order Roe flux scheme. This figure shows the error between each of the meshes and the finest mesh (2187×2187 cells). The meshes are refined by a factor of 3, so the error for a first-order scheme should decrease by a factor of 3 for each level of refinement. This occurs only for the finest meshes, so we may just be entering the asymptotic range.

Figure 4.5 shows the error versus time for the second-order Roe flux scheme. The error between the finer meshes decreases by a factor or 9, as would be expected of a second-order scheme when the mesh is refined by a factor of 3. This is observed for several levels of refinement, demonstrating that we are in the asymptotic range. Figure 4.6 shows the error versus time for the second-order Steger-Warming flux scheme, and Figure 4.7 shows the error versus time for the second-order Subbareddy-Candler scheme. Both schemes also exhibit a $9\times$ decrease in error for the finer meshes.

Figure 4.8 shows the error versus time for the fourth-order Subbareddy-Candler flux scheme. For this case, we would expect a $81\times$ decrease in error as the mesh is refined by a factor of 3. This level of decrease is observed initially for the finer meshes, but for most of the simulation time only a $9\times$ decrease in error is observed. The current hypothesis is that the issue has to do with the

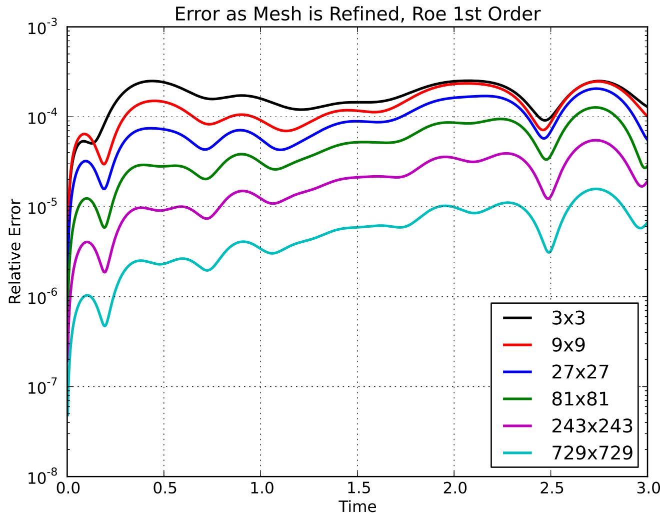


Figure 4.4. The relative errors between the solution on each mesh and the solution on the finest mesh for the inviscid pulse test case. The results shown are for the first-order Roe flux scheme in `SPARC`.

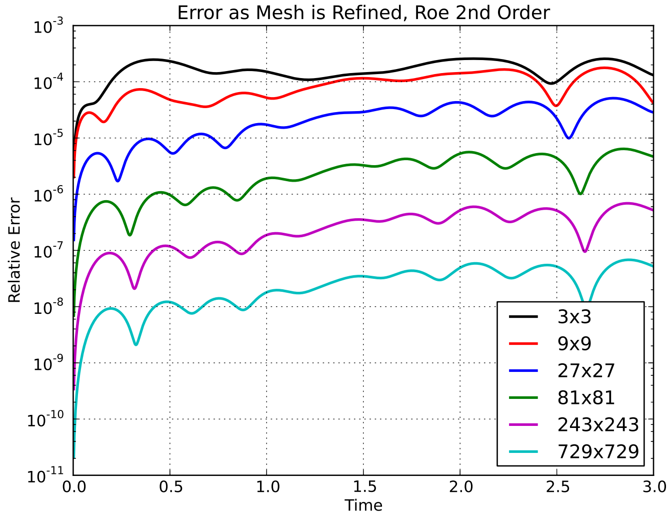


Figure 4.5. The relative errors between the solution on each mesh and the solution on the finest mesh for the inviscid pulse test case. The results shown are for the second-order Roe flux scheme in `SPARC`.

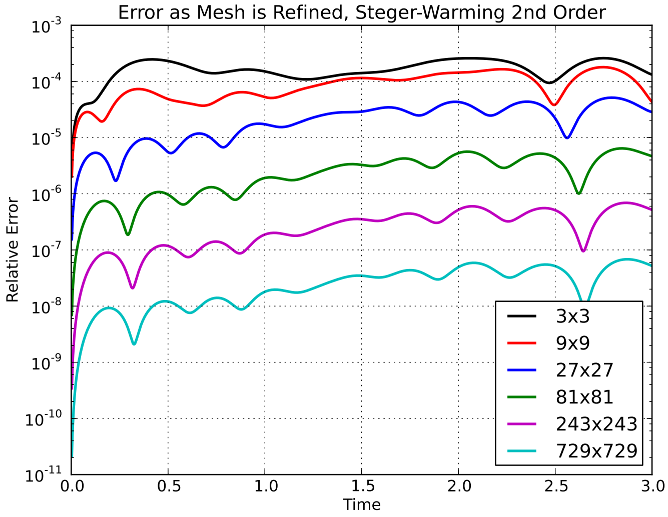


Figure 4.6. The relative errors between the solution on each mesh and the solution on the finest mesh for the inviscid pulse test case. The results shown are for the second-order Steger-Warming flux scheme in `SPARC`.

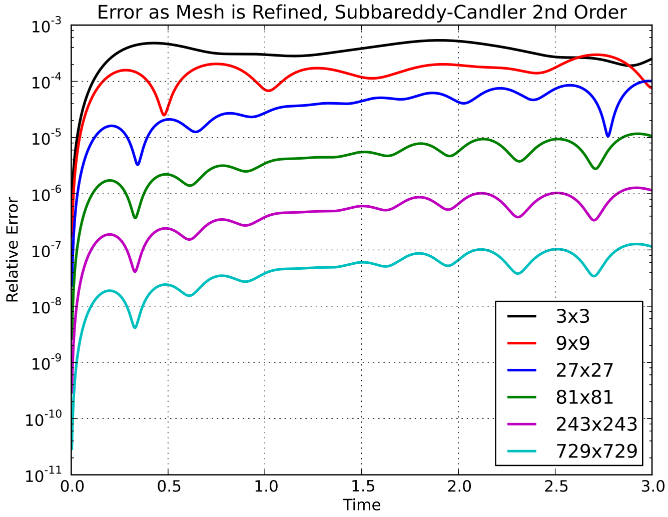


Figure 4.7. The relative errors between the solution on each mesh and the solution on the finest mesh for the inviscid pulse test case. The results shown are for the second-order Subbareddy-Candler flux scheme in `SPARC`.

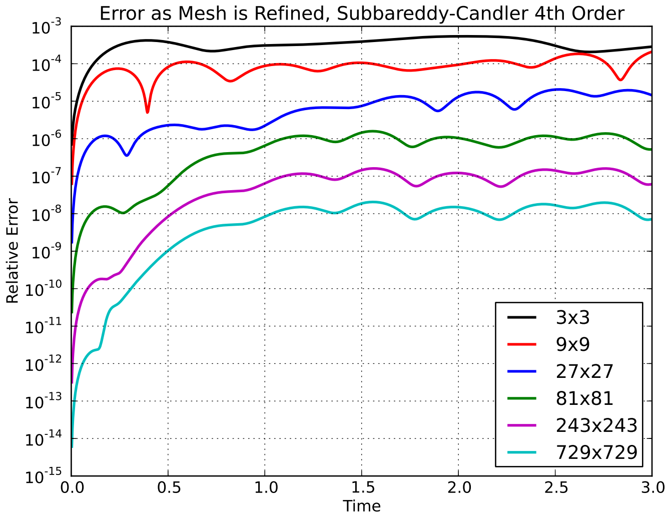


Figure 4.8. The relative errors between the solution on each mesh and the solution on the finest mesh for the inviscid pulse test case. The results shown are for the fourth-order Subbareddy-Candler flux scheme in `SPARC`.

boundary conditions, which are not anticipated to support fourth-order accuracy in their current form. The reasoning behind this hypothesis has to do with the degradation from fourth to second order at around the time when the pulse starts to interact with the solid walls.

Figure 4.9 shows the error plotted versus mesh size for the various flux schemes. The slope of the curves gives the convergence rate of the methods. Figure 4.10 shows the convergence rate computed using the finest meshes as a function of time. This plot verifies that the various schemes exhibit the stated order of accuracy, except for the fourth-order Subbareddy-Candler which is only second-order accurate.

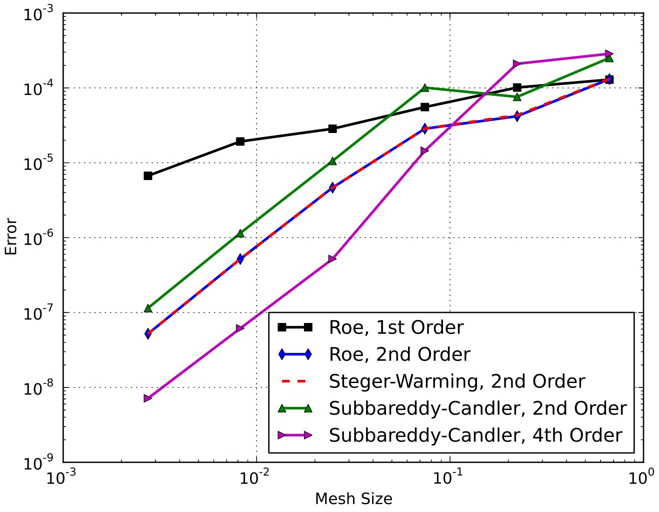


Figure 4.9. The relative errors for each flux scheme as the mesh size is varied.

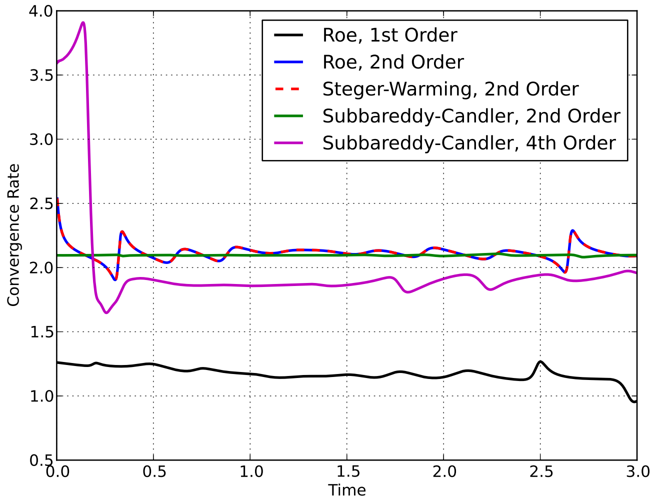


Figure 4.10. The convergence rates as a function of time for each flux scheme.

Chapter 5

Structure preservation in finite volume ROMs via physics-based constraints

5.1 Least-Squares Petrov-Galerkin model reduction formulation with physics-based constraints

In Computational Fluid Dynamics (CFD), it is critical that numerically-computed solutions respect the underlying flow physics, e.g., global conservation laws, energy or entropy stability, etc. While most viable CFD discretization approaches are developed to respect *a priori* at least some of these physical properties of fluid flows, projection-based ROMs constructed from these high-fidelity simulations are in general not guaranteed to preserve the same properties. For instance, a POD ROM obtained by projecting a conservative finite volume-based full order model may not be conservative.

One way to ensure that a projection-based ROM formulation respects certain physical properties of the underlying flow physics and/or full order discretization is through the introduction of physics-based constraints, an approach termed “structure preservation” [39]. Since LSPG-based ROMs (Chapter 2) are cast in a nonlinear least-squares framework, they are particularly amenable to a formulation that includes constraints, added in order to preserve important physical properties of the governing problem.

Here, we describe several structure-preserving constraints that are physically meaningful for compressible flow problems[1].

5.1.1 Global conservation laws

Consider the original FOM nonlinear algebraic equations (2.1) as corresponding to the solution increment at time step n:

$$\boldsymbol{r}^n(\boldsymbol{w}^n) = 0, \quad n = 1, \ldots, T \tag{5.1}$$

---

[1] Structure-preservation for other application is also possible; see [9] for a model reduction methodology that preserves Lagrangian structure intrinsic to mechanical systems, and [34] for a structure-preserving model reduction methodology for marginally-stable linear time-invariant systems.

and the associated ROM solution $\tilde{\boldsymbol{w}}^n = \bar{\boldsymbol{w}} + \boldsymbol{\Phi}_M \hat{\boldsymbol{w}}^n$, where $\bar{\boldsymbol{w}} = \boldsymbol{w}^0$ typically. Assume the FOM from which (5.1) was derived was a finite volume discretization of a governing system of fluid equations. Then, a reasonable set of constraints for (5.1) are global conservation laws, that is, the mass, momentum and energy conservation expressions integrated over the problem domain Ω.

Let ρ denote the fluid density, u_i denote the i^{th} component of the fluid velocity for $i = 1, \dots 3$ in three spatial dimensions, p denote the fluid pressure, e denote the fluid energy and $\boldsymbol{x}$ denote the spatial coordinate vector. First, the global continuity equation can be written as

$$\begin{aligned}\int_{t^{n-1}}^{t^n} \int_{\Gamma} u_i(\boldsymbol{x},\tau)\rho(\boldsymbol{x},\tau)n_i(\boldsymbol{x})dSd\tau &= \int_{\Omega} \rho(\boldsymbol{x},t^n)d\Omega - \int_{\Omega} \rho(\boldsymbol{x},t^{n-1})d\Omega && (5.2)\\ &= \boldsymbol{a}_{\rho}^T(\tilde{\boldsymbol{w}}^n - \tilde{\boldsymbol{w}}^{n-1}) && (5.3)\\ &= \boldsymbol{a}_{\rho}^T\boldsymbol{\Phi}_M(\hat{\boldsymbol{w}}^n - \hat{\boldsymbol{w}}^{n-1}), && (5.4)\end{aligned}$$

where we have used index notation and $\boldsymbol{a}_{\rho} \in \mathbb{R}^N$ is a vector with all zeros except for the degrees of freedom corresponding to the density degrees of freedom; for these degrees of freedom, the value of the entry of $\boldsymbol{a}_{\rho}$ is the volume of the finite-volume cell. We use the symbol Γ to denote the boundary of the domain Ω, and n_i to denote the i^{th} component of the normal vector to Γ, for $i = 1, \dots 3$ in three spatial dimensions.

Analogously, we can write the three global momentum conservation laws as

$$\begin{aligned}\int_{t^{n-1}}^{t^n} \int_{\Gamma} (u_i(\boldsymbol{x},\tau)\rho(\boldsymbol{x},\tau)u_j(\boldsymbol{x},\tau)n_i(\boldsymbol{x}) + p(\boldsymbol{x},\tau)n_j(\boldsymbol{x}))dSd\tau &= \int_{\Omega} \rho(\boldsymbol{x},t^n)u_j(\boldsymbol{x},t^n)d\Omega && (5.5)\\ &\quad - \int_{\Omega} \rho(\boldsymbol{x},t^{n-1})u_j(\boldsymbol{x},t^{n-1})d\Omega, && (5.6)\\ &= \boldsymbol{a}_{u_j}^T(\tilde{\boldsymbol{w}}^n - \tilde{\boldsymbol{w}}^{n-1}), && (5.7)\\ &= \boldsymbol{a}_{u_j}^T\boldsymbol{\Phi}_M(\hat{\boldsymbol{w}}^n - \hat{\boldsymbol{w}}^{n-1}), && (5.8)\end{aligned}$$

for $j = 1, \dots, 3$ and the global energy conservation law as

$$\begin{aligned}\int_{t^{n-1}}^{t^n} \int_{\Gamma} u_i(\boldsymbol{x},\tau)[\rho(\boldsymbol{x},\tau)e(\boldsymbol{x},\tau) + p(\boldsymbol{x},\tau)]n_i(\boldsymbol{x})dSd\tau &= \int_{\Omega} \rho(\boldsymbol{x},t^n)e(\boldsymbol{x},t^n)d\Omega && (5.9)\\ &\quad - \int_{\Omega} \rho(x,t^{n-1})e(\boldsymbol{x},t^{n-1})d\Omega && (5.10)\\ &= \boldsymbol{a}_{e}^T(\tilde{\boldsymbol{w}}^n - \tilde{\boldsymbol{w}}^{n-1}) && (5.11)\\ &= \boldsymbol{a}_{e}^T\boldsymbol{\Phi}_M(\hat{\boldsymbol{w}}^n - \hat{\boldsymbol{w}}^{n-1}). && (5.12)\end{aligned}$$

For a more detailed discussion of the POD/LSPG ROM formulation with global conservation law constraints, the reader is referred to [39].

5.1.2 Clausius-Duhem inequality (second law of thermodynamics)

In addition to the conservation law constraints described in Section 5.1.1, one may also consider adding as a constraint the Clausius-Duhem inequality, a statement of the second law of thermodynamics. Solutions satisfying this inequality are typically referred to as “entropy-stable”. Entropy stability is an important physical property for several reasons, namely it implies nonlinear stability and entropy-stable discretization methods typically require less numerical dissipation.

For the dimensionless, compressible Navier-Stokes equations, the weak integral form of Clausius-Duhem inequality states that:

$$\frac{d}{dt}\int_{\Omega}\rho\eta d\Omega \geq \int_{\partial\Omega}\rho\eta(u_n - \mathbf{v}\cdot\mathbf{n})dS - \int_{\Gamma}\frac{q_i n_i}{\theta}dS + \int_{\Omega}\frac{\rho s_e}{\theta}d\Omega - \int_{\Omega}\frac{\rho u s_{\rho u}}{\theta}d\Omega. \tag{5.13}$$

Here, ρ denotes the fluid density, η denotes the thermodynamic entropy density per unit mass, θ denotes the absolute temperature of the fluid, q_i denotes the heat flux, u_n is the normal velocity of $\partial\Omega$, $\mathbf{v}$ is the velocity of particles inside Ω, $\mathbf{n}$ is the unit normal to $\partial\Omega$, s_e is an energy source per unit mass, and $s_{\rho u}$ is a momentum source term per unit mass. In the specific case where $s_{\rho u} = s_e = 0$ (no energy and momentum source terms), $u_n = 0$ (the domain Ω is fixed) and the flow satisfies a no-slip or no-penetration BC on $\partial\Omega$, (5.13) reduces to:

$$\frac{d}{dt}\int_{\Omega}\rho\eta d\Omega \geq -\int_{\partial\Omega}\frac{q_i n_i}{\theta}dS. \tag{5.14}$$

Effectively, (5.14) states that the entropy of the fluid is non-decreasing, a physical property of fluid flows. It was shown in [25] that the physics dictate that a numerically computed solution to the compressible Navier-Stokes equations must satisfy the Clausius-Duhem inequality (5.14). Satisfaction of (5.14) by a numerical solution was termed “entropy-stability”. Integrating (5.14) over time, we obtain:

$$\int_{\Omega}\rho(x,t^n)\eta(x,t^n)d\Omega - \int_{\Omega}\rho(x,t^{n-1})\eta(x,t^{n-1})d\Omega + \int_{t^{n-1}}^{t^n}\int_{\partial\Omega}\frac{q_i(x,\tau)n_i}{\theta(x,\tau)}dSd\tau. \tag{5.15}$$

Our task here is to rewrite (5.14) using the usual fluid variables, ρ, u_i and e. From [25], one has the following identities:

- Relation between non-dimensional entropy s and other fluid variables:

  $$s \equiv \frac{\eta}{c_v} = \ln\left(p\rho^{-\gamma}\right), \tag{5.16}$$

  where c_v is the specific pressure at constant volume, p is the fluid pressure, and γ is the ratio of specific heats.

- Ideal gas law, which relates pressure to density and energy, and temperature to energy:

  $$p = (\gamma-1)\rho e, \quad \theta = \gamma(\gamma-1)M_{ref}^2 e, \tag{5.17}$$

  where M_{ref} is the Mach number of the flow.

- Definition of the heat flux:

  $$q_i = -\kappa\theta_{,i} \tag{5.18}$$

  where κ is the thermal conductivity.

Using (5.16) and (5.17), we have that

$$\eta = c_v \ln\left[(\gamma-1)\rho^{1-\gamma} e\right]. \tag{5.19}$$

Using (5.17) and (5.18), we also have that

$$\frac{q_i n_i}{\theta} = -\frac{\kappa e_{,i} n_i}{e} = -\kappa\left(\frac{\nabla(\rho e)}{\rho e} - \frac{\nabla\rho}{\rho}\right)\cdot\mathbf{n}. \tag{5.20}$$

Substituting (5.20) and (5.19) into (5.15), we obtain the following:

$$\begin{aligned}&\textstyle\int_\Omega c_v \rho(x,t^n)\ln\left[(\gamma-1)(\rho e)(x,t^n)\rho(x,t^n)^\gamma\right] d\Omega - \int_\Omega \rho(x,t^{n-1})\ln\left[(\gamma-1)(\rho e)(x,t^{n-1})\rho(x,t^{n-1})^\gamma\right] d\Omega \\ &\textstyle - \int_{t^{n-1}}^{t^n}\int_{\partial\Omega}\kappa\left(\frac{\nabla(\rho e)(x,\tau)}{(\rho e)(x,\tau)} - \frac{\nabla\rho(x,\tau)}{\rho(x,\tau)}\right)\cdot\mathbf{n}\, dS d\tau \geq 0.\end{aligned} \tag{5.21}$$

5.1.3 Total variation diminishing (TVD) and total variation bounded (TVB) properties

Another inequality constraint to consider adding to the ROM is a total variation diminishing (TVD) constraint. The total variation (TV) of a scalar quantity U in the discrete one-dimensional (1D) case is defined as

$$TV(U) = \sum_j |U_{j+1} - U_j| \tag{5.22}$$

(here U_j denotes the value of U at grid-point j). Equation (5.22) approximates the continuous definition of the total variation as the L_1 norm of the gradient, namely $\int_\Omega |\frac{\partial U}{\partial x}| d\Omega$ for some domain Ω.

A TVD constraint enforces that $TV(U^{n+1}) \leq TV(U^n)$, where n denotes the time step. This constraint is utilized to avoid the nonphysical and spurious oscillations that can occur in numerical schemes. Due to high frequency modes for a low dimensional basis, LSPG ROMs can become oscillatory even if the full order model is not. The TVD constraint is of especial importance in modeling problems with shocks in hyperbolic conservation laws.

For realistic problems, it is useful to extend the definition of TVD (5.22) beyond the 1D case. The total variation definition easily extends to higher dimensions. For example,

$$\begin{aligned}TV(U) = &\limsup_{\varepsilon\to 0}\frac{1}{\varepsilon}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}|U(x+\varepsilon,y) - U(x,y)|dxdy \\ &+\limsup_{\varepsilon\to 0}\frac{1}{\varepsilon}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}|U(x,y+\varepsilon) - U(x,y)|dxdy\end{aligned} \tag{5.23}$$

in the two dimensional case [28], where x and y denote the coordinate dimensions. This leads to the corresponding discrete form

$$TV(U) = \sum_{i=-\infty}^{\infty}\sum_{j=-\infty}^{\infty}[|U_{i+1,j} - U_{i,j}|\Delta y + [|U_{i,j+1} - U_{i,j}|]\Delta x.$$

For hyperbolic systems, such as the compressible Euler equations of gas dynamics, the total variation can be defined component-wise or in terms of the characteristic variables. Unlike in the nonlinear scalar conservation law case, the total variation in x of the solution in the system case is not guaranteed to be a monotonically non-increasing function of time due to nonlinear wave interactions [46]. It can be shown that the total variation with respect to the characteristic variables is decreasing in the constant coefficient case, where the system reduces to m scalar nonlinear conservation laws [19]. Let $\boldsymbol{A}(\boldsymbol{U})$ be the diagonalizable Jacobian matrix with $\boldsymbol{a}^l(\boldsymbol{U})$ eigenvalues and $\boldsymbol{R}^l(\boldsymbol{U})$ eigenvectors, so $\boldsymbol{R}^{-1}\boldsymbol{A}\boldsymbol{R} = diag(\boldsymbol{a}^l)$, where $l = 1, \cdots, m$. Then $\boldsymbol{W} = \boldsymbol{R}^{-1}\boldsymbol{U}$ and the total variation is then defined as

$$TV(\boldsymbol{U}) = \sum_{j=-\infty}^{\infty} \sum_{l=1}^{m} |\boldsymbol{w}_{j+1}^{l} - \boldsymbol{w}_{j}^{l}|, \tag{5.24}$$

where $\boldsymbol{w}_{j+1}^{l} - \boldsymbol{w}_{j}^{l}$ is the component of $\boldsymbol{U}_{j+1} - \boldsymbol{U}_j$ in the l^{th} characteristic direction [18].

For hyperbolic systems of equations, total variation bounded (TVB) may also be worth considering. TVB constraints are weaker than TVD constraints because they allow the total variation of the solution to increase by a specified bound β, that is we require that

$$TV(\boldsymbol{U}) \leq \beta.$$

The TVB approach involves parameter tuning to find the optimal β. Machine learning methods could be incorporated to learn the β parameter from the snapshots.

5.1.4 Rotational quantities

Conservation of rotational quantities is also important in modeling problems with turbulent flow and vortices. Let $\omega = \nabla \times u$ refer to the vorticity, where u is the velocity. We can then define the enstrophy and helicity. The enstrophy is given as the L_2-norm of ω, namely

$$\int_{\Omega} |\omega|^2 d\Omega. \tag{5.25}$$

Enstrophy is conserved for inviscid problems in two dimensions [37, 13]. The vortex tube stretching term $(\omega \cdot \nabla)u$ does not vanish in three dimensional flow and so enstrophy is not conserved in three dimensions [16]. Similar to the techniques for proving nonlinear stability using the entropy norm, one can also show nonlinear stability using energy and enstrophy norms.

While enstrophy is not conserved for three-dimensional flows, helicity defined as

$$\int_{\Omega} u \cdot \omega d\Omega \tag{5.26}$$

is conserved. Helicity is a metric to quantify how the velocity lines coil around each other [37]. Implementation of these rotational equality constraints follows similarly to the global conservation equality constraints.

5.1.5 Discussion

In formulating a ROM with constraints, it is important to note that the number of constraints cannot exceed the number of degrees of freedom M in the reduced-order model, or else the problem becomes overdetermined. The number of constraints can be increased by decomposing the domain Ω into a set of subdomains and enforcing the constraints over each subdomain rather than over the entire domain Ω, as long as the problem does not become overdetermined.

Another important consideration in formulating ROMs with structure-preserving constraints is consistency. In particular, if one introduces constraints into a ROM that the FOM solution is not guaranteed to satisfy, one runs the risk of introducing an inconsistency between the full and reduced models. In general, finite volume schemes for fluid problems are, by construction, conservative, meaning the global conservation law constraints described in Section 5.1.1 will be satisfied by the FOM solution if it is computed using a finite volume scheme. Satisfaction of the Clausius-Duhem inequality (Section 5.1.2) in general requires a transformation of the governing equations into the so-called “entropy variables” [25, 22, 17]. Although some entropy-stable (Clausius-Duhem-satisfying) formulations lack the global conservation property, there do exist entropy-stable conservative finite volume and Discontinuous Galerkin (DG) methods, e.g., [15] and [10] respectively. Similarly, for enstrophy preserving schemes, the governing equation is written in terms of the vorticity [44, 33]. Numerical schemes with slope limiters result in high resolution and non-oscillatory TVD solutions [23].

Note that many FOM formulations can in principle be modified to ensure the satisfaction of various desired physical properties, e.g., through the introduction of constraints in an optimization-like formulation [5]. This observation may justify the prescription of FOM-inconsistent constraints in a ROM formulation, on the grounds that the FOM can be modified such that it too satisfies the desired constraints.

5.2 Demonstration on 1D conservation laws

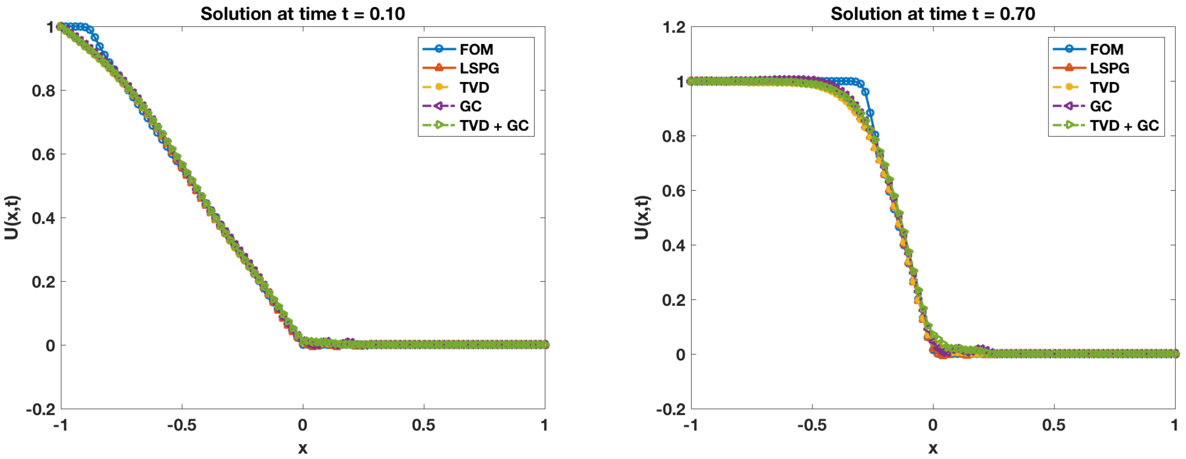
5.2.1 Burger’s equation

We first demonstrate the effect of the global conservation and TVD constraints on a one-dimensional scalar conservation law. We use Burger’s equation as the governing equation,

$$u_t + f(u)_x = 0, \tag{5.27}$$

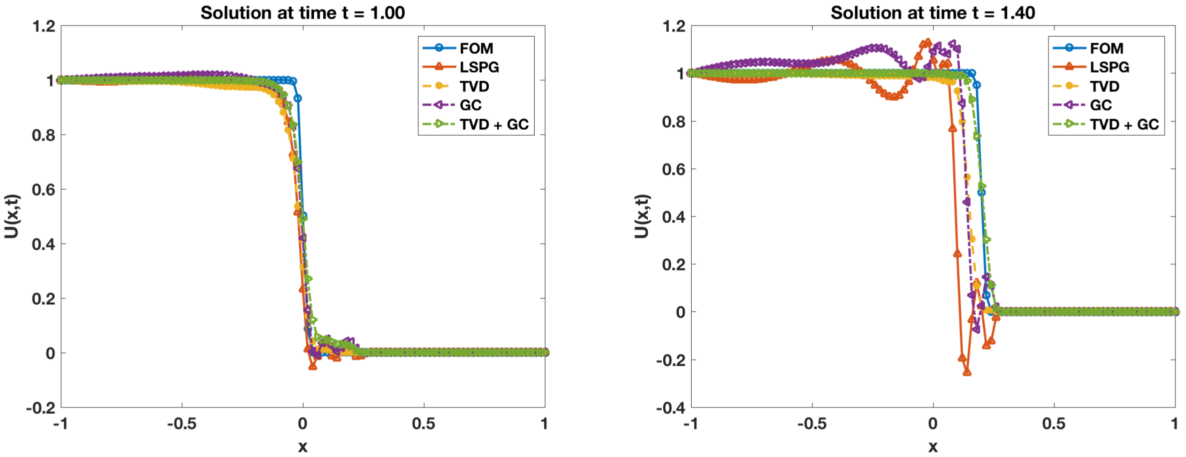
where $f(u) = \frac{u^2}{2}$ is a convex nonlinear flux function.

The numerical results of the reproductive ROM with basis dimension $nD = 10$, spatial step size $\Delta x = 0.02$ and $\Delta t = \Delta x^2$ are shown in Figures 5.1–5.2. The FOM is generated using a symmetric limited positive (SLIP) scheme with superbee limiter [23] implemented in `MATLAB`. 3751 snapshots were generated and the percent energy captured by the $nD = 10$ basis is 91%.


Solution at time t = 0.10

Solution at time t = 0.70

Figure 5.1. Spatial profile as the solution evolves with time.


Solution at time t = 1.00

Solution at time t = 1.40

Figure 5.2. Spatial profile as the shock forms and evolves with time.

Figure 5.2 illustrates that the oscillations with the unconstrained LSPG and LSPG with global conservation constraint amplify when the shock is formed and evolves. The global conservation constraint decreases the amplitudes of these oscillations, whereas the TVD constraints removes the oscillations. The solutions with unconstrained LSPG and LSPG with global conservation constraint are also non-physical since the maximum and minimum principles for this scalar nonlinear conservation law are violated.

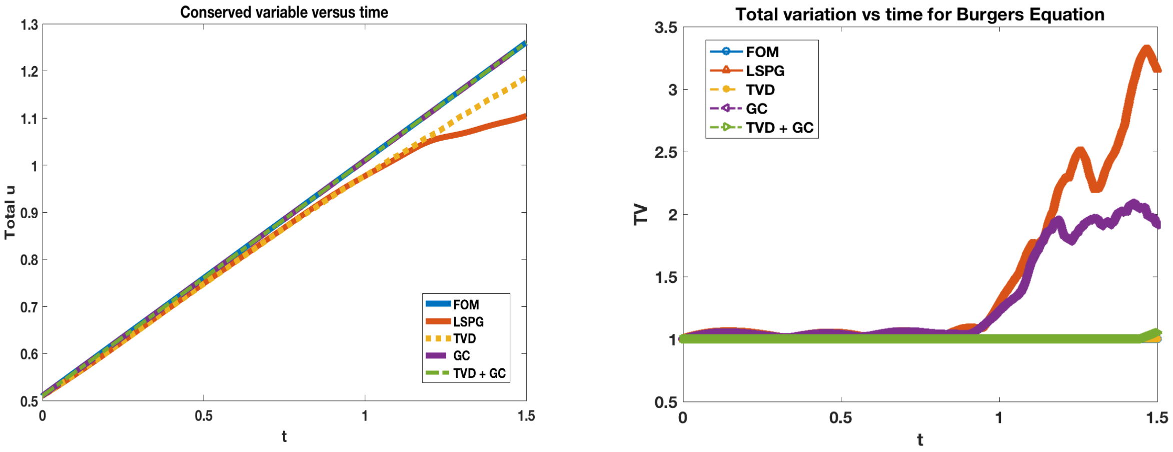


Figure 5.3. Conserved variable and total variation as functions of time.

Figure 5.3 depicts that even though conservation is satisfied in the finite volume FOM, conservation is not preserved in the unconstrained ROM. The ROM with global conservation constraint preserves the structure. Note that by integrating the one-dimensional Burger's equation in space and time, the global conservation constraint reduces to

$$\int_{\Omega} u(x,t^n)d\Omega - \int_{\Omega} u(x,t^{n-1})d\Omega = -\int_{t^{n-1}}^{t^n} f(u_N,t) - f(u_1,t)dt. \tag{5.28}$$

where $u_N = u(x_N,t)$ and $u_1 = u(x_1,t)$. The conservation error in the ROM with TVD constraint is improved, but the scheme is still not conservative. Similarly, Figure 5.3 also depicts that the total variation is improved in the ROM with the global conservation constraint, but matches the true solution only when the TVD constraint is used. For both properties to be satisfied, both constraints must be enforced.

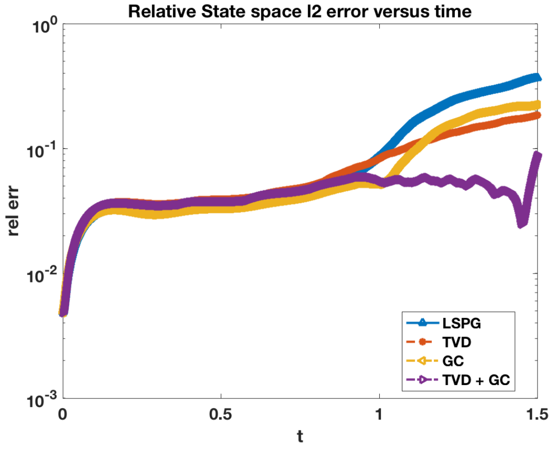


Figure 5.4. Relative state-space error and total variation as functions of time.

Figure 5.4 reveals that the relative state space error, as measured in the discrete l_2-norm is decreased the constrained ROM cases. The minimal error occurs when both TVD and global conservation are enforced.

5.2.2 Sod's shock tube problem

The governing equations for Sod's shock tube problem are the one-dimensional compressible Euler equations, given by the following nonlinear systems of conservation laws

$$\mathbf{u}_t + \mathbf{f}(\mathbf{u})_x = 0, \tag{5.29}$$

where $\mathbf{u} = \begin{pmatrix} \rho, & \rho u, & E \end{pmatrix}^T$ and $\mathbf{f}(\mathbf{u}) = u\mathbf{u} + \begin{pmatrix} 0, & p, & up \end{pmatrix}^T$, with the variables ρ, u and E denoting the fluid density, velocity and energy, respectively. The initial density and pressure are discontinuous and are given by

$$\rho^0(x) = \begin{cases} 1 & \text{if } x < 0.5, \\ 0.125 & \text{otherwise,} \end{cases}$$

and

$$p^0(x) = \begin{cases} 1 & \text{if } x < 0.5, \\ 0.1 & \text{otherwise,} \end{cases}$$

for $x \in \Omega = [0,1]$. The initial velocity is uniformly zero [41].

The FOM is a second order finite volume method with Roe flux and entropy fix from the `MORTestbed MATLAB` code [47]. A predictive LSPG ROM is implemented and the results are compared from the unconstrained and TVD constrained solutions. The training is run until time $t_{Train} = 0.25$. In these results, we test the long-time behavior of the ROM for times past the training window. The simulations are run with basis dimension $nD = 10$, spatial step size $\Delta x = 0.01$ and temporal step size $\Delta t = 0.001$. 1000 snapshots were used to generate the ROM and the percent energy captured by the $nD = 10$ basis is 99%.

Since the governing equation is a nonlinear system of hyperbolic conservation laws, the total variation of the solution is not guaranteed to be non-increasing, as discussed in Section 5.1.3. For consistency, we impose the TVD constraints on the solution conserved variables that satisfy the TVD property in the FOM. Figure 5.10 shows that the total variation of density and energy are constant in the FOM, whereas the total variation of the momentum is increasing. Hence, we impose the TVD constraints component-wise on the density and momentum.

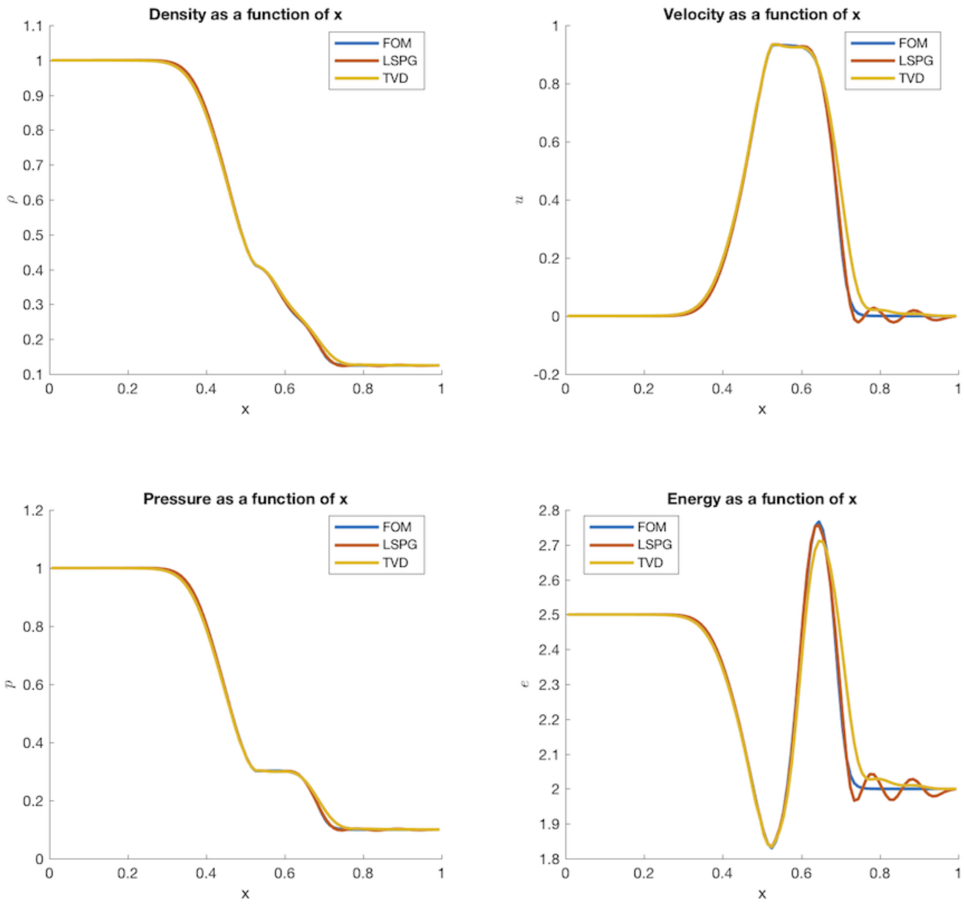


Figure 5.5. Spatial profile at time $t_{ROM final} = 0.1$.

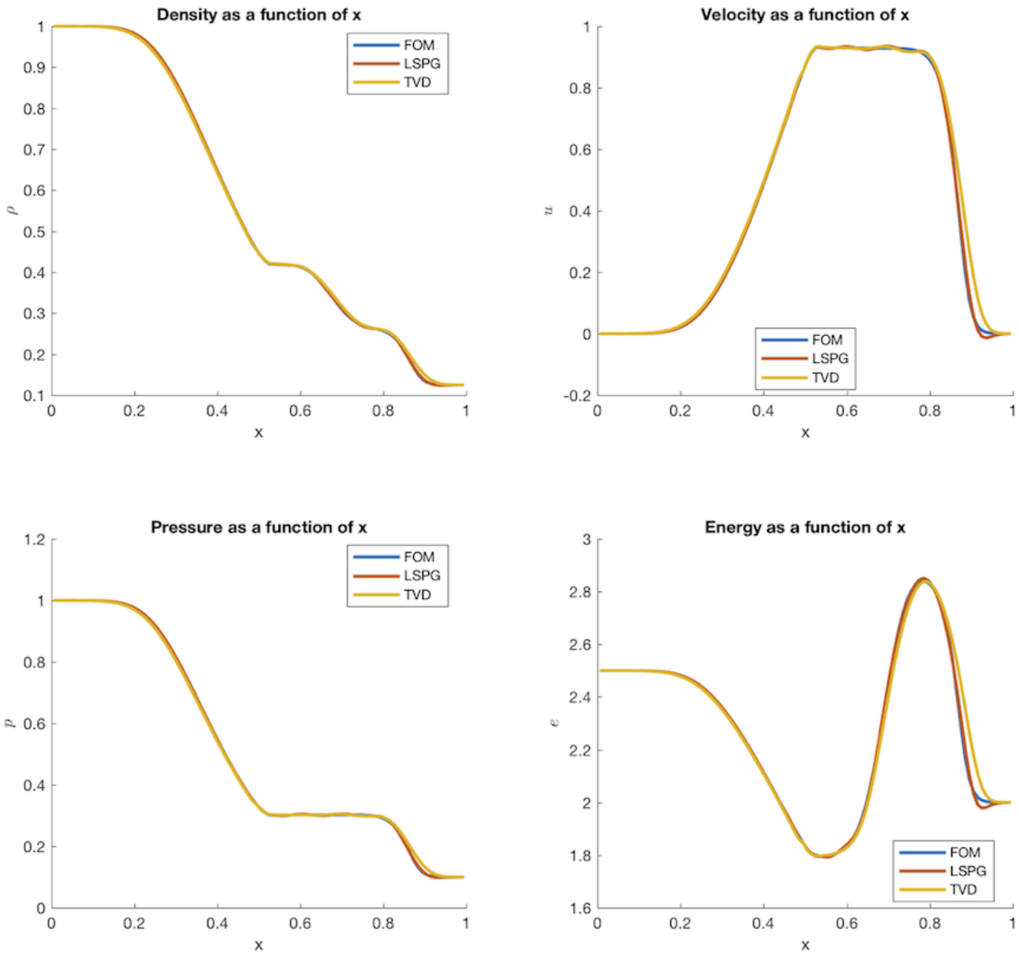


Figure 5.6. Spatial profile at time $t_{ROMfinal} = 0.2$.

Figures 5.5 and 5.6 display the numerical solutions at times within the training window. There are low amplitude oscillations present in the unconstrained LSPG solution in Figure 5.5 and a nonphysical negative velocity present in this solution in Figures 5.5 and 5.6. These numerical artefacts are removed in the TVD constrained LSPG ROM.

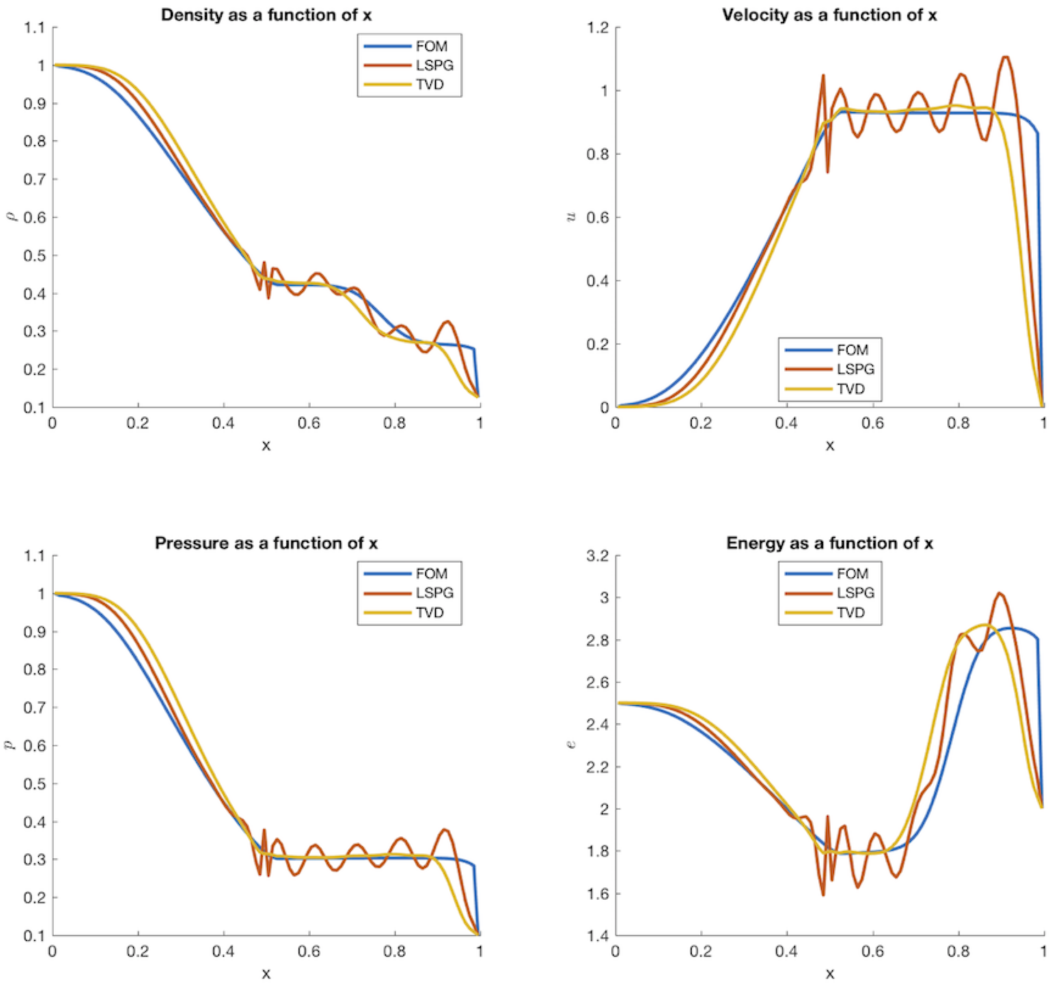


Figure 5.7. Spatial profile at time $t_{ROM final} = 0.3$.

Figures 5.7 and 5.8 show that the LSPG ROM becomes highly oscillatory and unstable for times outside of the training window. The TVD constrained solution removes the nonphysical oscillations and provides a stable solution for shock problems.

Figure 5.9 reveals that while global conservation holds for the total mass and total mass, the ROM does not satisfy conservation of total momentum. Future work will include combining the global conservation equality constraints with the TVD inequality constraint as done in Burger's equation [30].

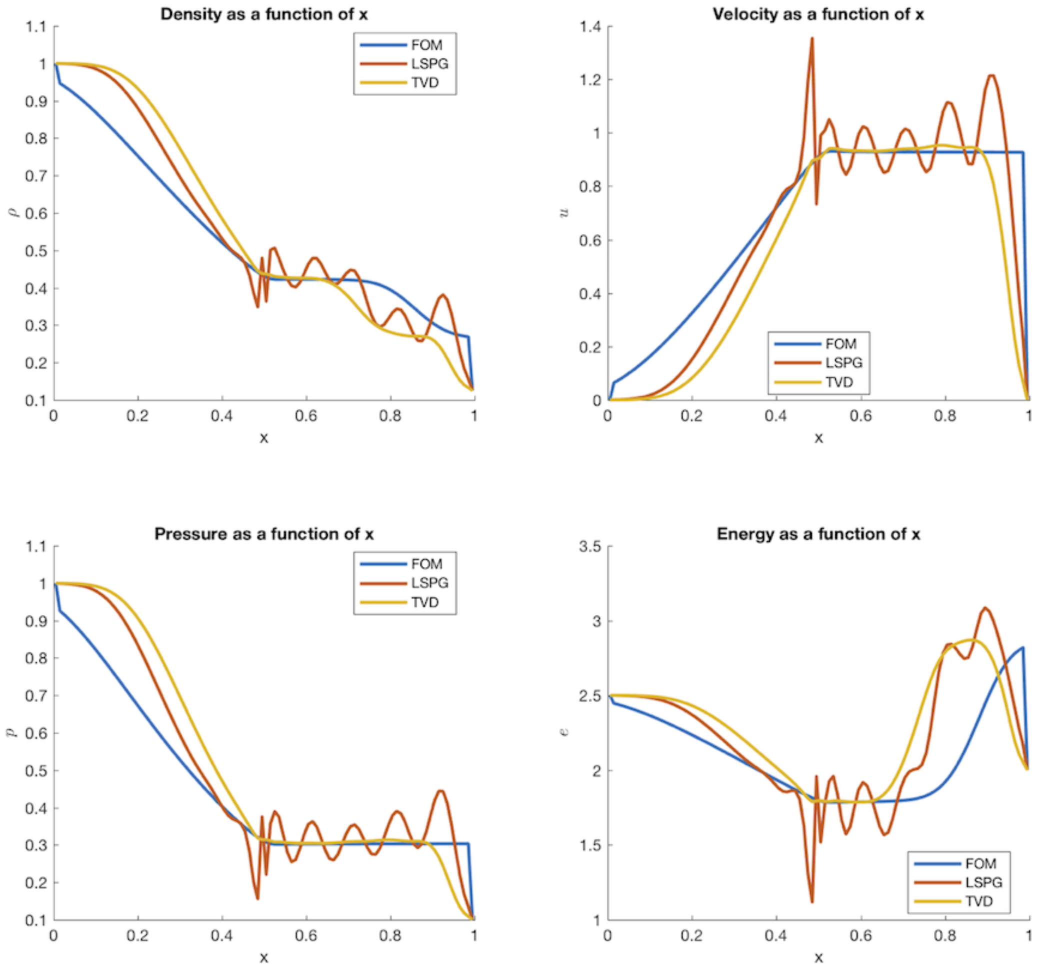


Figure 5.8. Spatial profile at time $t_{ROMfinal} = 0.4$.

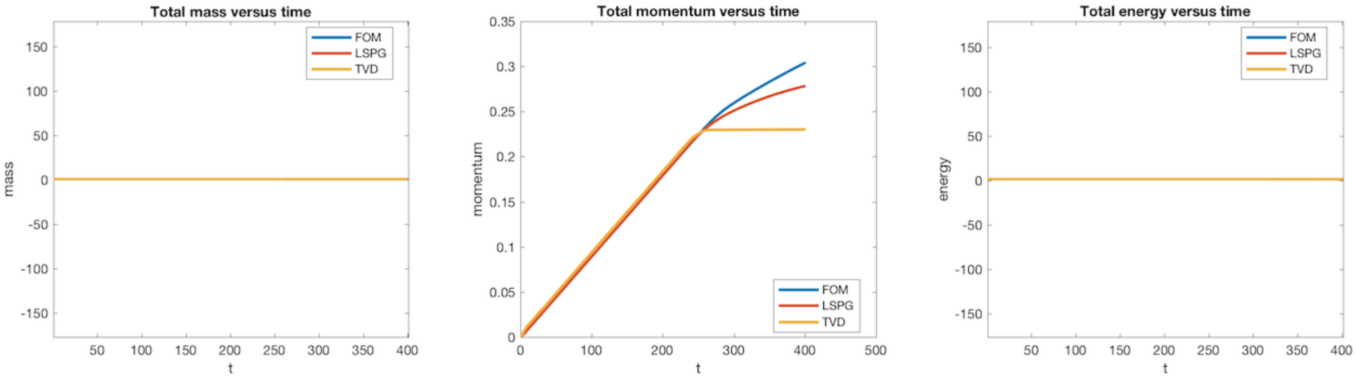


Figure 5.9. Total conserved variables as a function of time.

The total variation of all the conserved variables increases in the unconstrained LSPG ROM above the total variation of the conserved variables in the ROM. Imposing the TVD constraints in the ROM results in total variations of the conserved variables consistent with those of the FOM.

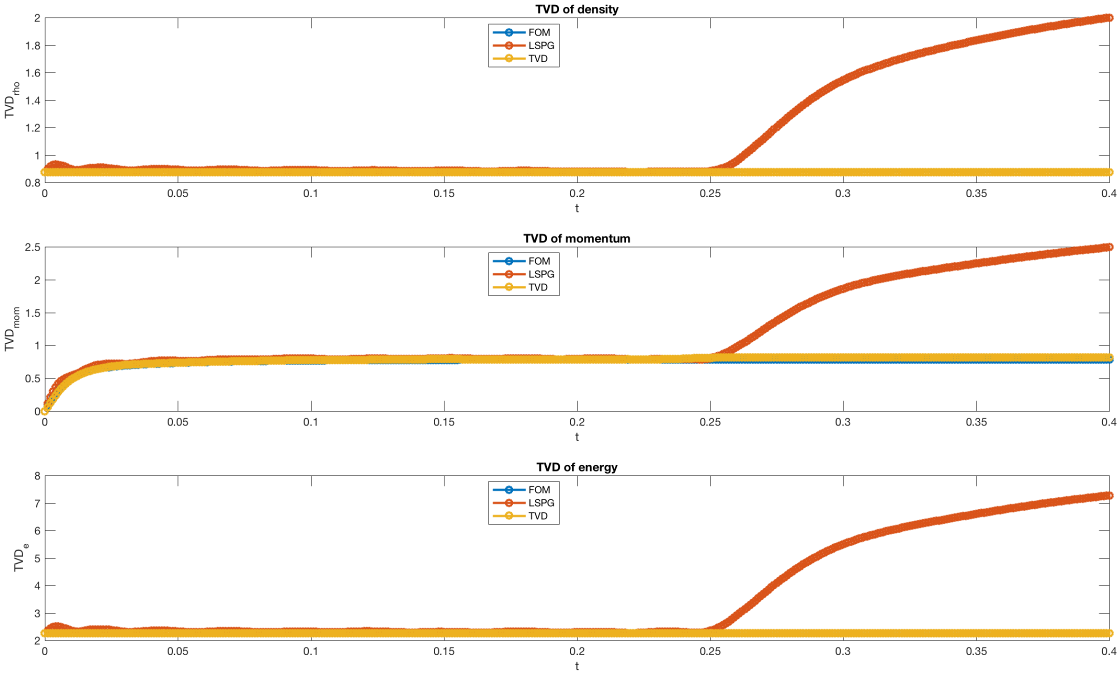


Figure 5.10. Total variation as a function of time

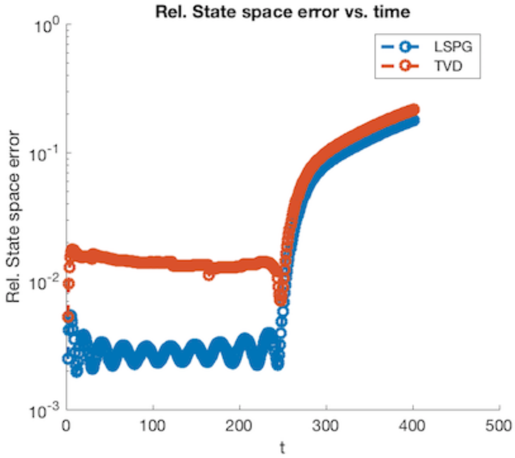


Figure 5.11. Relative state space error as a function of time.

In Figure 5.11, the relative state space discrete l_2 error is larger in the TVD constrained ROM, even though it provides non-oscillatory solutions. The difference in the error is more significant at early times. The loss of accuracy of the TVD constrained solution can be attributed to the fact that with a low-dimensional basis to find a feasible point, the least squares residual may not be as decreased as in the unconstrained case.

To improve the accuracy of TVD constrained solution, future work includes experimenting with different bases, such as using the snapshots basis with a sparse l_1 penalty. Another approach is to partition domain into k different regions and compute a basis for each region. We may also conjecture that adding the global conservation constraint will decrease the relative state-space error, as observed with Burger's equation.

5.3 Perspectives towards `SPARC` implementation

The results summarized above show that it is possible to decrease the relative state space error and preserve the physical structure of a ROM solution through the introduction of various physics-based constraints into the LSPG/POD ROM formulation. These promising preliminary results suggest that the addition of constraints into the `SPARC` ROM formulation may be beneficial. As discussed in our FY16 report [45] and Section 3.4.4, scenarios in which `SPARC` LSPG/POD ROMs fail by computing non-physical quantities (e.g., negative pressures) do occur, which suggests some physics-based property is being violated by these models. Prior to adding a constrained LSPG/POD ROM formulation into `SPARC` it is worthwhile to perform a numerical study of whether these ROMs satisfy the candidate constraints (e.g., conservation).

The following constraints would be worthwhile to consider adding to the unconstrained LSPG/POD formulation in `SPARC`:

- *Global conservation.* We have seen in the above examples that even if the FOM is conservative, the ROM is not necessarily. The high-fidelity finite volume schemes employed in `SPARC` are, by construction, conservative. Enforcing the global conservation constraint results will result in a physically correct conservative ROM.

- *Clausius-Duhem inequality.* Similarly to global conservation, enforcing the Clausius-Duhem inequality in model will result in an entropy preserving numerical solution. The implementation of high-fidelity entropy-stable conservative finite volume and DG methods, e.g., [15] and [10], in `SPARC` is planned for FY18.

- *Total variation boundedness (TVB).* This constraint will be relevant to `SPARC` problems that are supersonic and exhibit shocks. Stability and long time behavior of ROMs can be improved by bounding the total variation. The TVB constraint can mitigate the spurious oscillations, commonly found in problems with shocks.

- *Enstrophy and helicity conservation.* Enstrophy and helicity conservation constraints are important when modeling vortices and turbulence in the two and three dimensional flows. Note that enstrophy and helicity conservation often requires a special numerical formulation, so these quantities may not be conserved by `SPARC` FOMs for all problems.

Finally, it is worth noting that, while physics-based structure-preserving constraints can improve the accuracy of ROMs based on the POD method, the accuracy of these models will likely be limited by the quality and span of the reduced basis employed, especially for predictive problems. The combination of LSPG/POD ROMs with basis enhancement and/or enrichment methodologies, e.g., [6] and [2], is therefore recommended as a longer-term research endeavor.

Chapter 6

Summary and future work

6.1 Summary

This report summarizes our FY17 study of the viability of the LSPG/POD model reduction methodology implemented within the `SPARC` finite volume flow solver in the context of a 2D viscous laminar cavity flow problem. This problem is a simplified representation of the captive-carriage environment, of interest to Sandia National Laboratories for the design and qualification of NW systems. Of particular interest are long-time simulations, in which the ROM is run at the same parameters as the FOM from which it is constructed, but much longer in time.

The primary focus of this work is to see if reduced-order models can accurately reproduce the PSD of the pressure fluctuations from the full-order model. We find that ROM accuracy and stability can be improved by running the ROM non-dimensionally, varying the ROM training window, introducing a preconditioner into the ROM LSPG formulation that has the effect of modifying the norm in which the residual is minimized, and varying the timestep used in the ROM. Our results demonstrate that the improved ROMs can produce PSDs with peaks matching their corresponding FOMs, a promising result that warrants the continuation of this work in the future.

Also described in this report are the results of a formal mesh convergence study of various finite volume discretization methods within `SPARC`, and a structure-preserving constrained LSPG ROM formulation. The mesh convergence study demonstrates that the expected order of accuracy is in general achieved for the schemes in `SPARC` in the context of an inviscid pressure pulse problem. We show in the context of some 1D conservation laws that the constrained structure-preserving formulation makes it possible to decrease the relative state space error and preserve the physical structure of a ROM solution, which suggests that adding constraints to the LSPG/POD ROMs in `SPARC` in the future may be a worthwhile effort.

6.2 Future work

During FY17, significant progress was made in improving the predictive capabilities of the LSPG/POD ROMs through the work detailed in this report. However, further research is required to gain a complete picture of the viability of the proposed ROM methodology for the targeted application.

In addition, code optimizations are required to achieve the desired speedups (relative to a `SPARC` high-order model) by using the ROMs in `SPARC`.

The primary FY18 tasks for this project will center around: (***Objective 1***) gaining as complete as possible of an understanding of the viability of the LSPG/POD ROMs implemented in the `SPARC` flow solver in the specific case of a 2D viscous laminar cavity problem, and (***Objective 2***) improving the efficiency of the said ROMs so as to understand what speedups are possible with respect to a full order simulation, as well as to have a complete `SPARC` LSPG/POD ROM implementation that can be used by future researchers/analysts working with the code.

In addition to the two objectives identified above, we will complete and verify the implementation of the higher order Rai scheme [35] commenced during FY17. This is expected to benefit the Advanced Technology Development and Mitigation (ATDM) project and enable 3D cavity ROM simulations at a later point in time. Specific tasks and measurable outcomes for each of these two primary objectives identified above are discussed below.

Towards ***Objective 1***, we will focus on gaining a complete understanding of the effect of the following "tunings" involving our `SPARC` LSPG/POD ROMs, all in the context of a 2D viscous laminar cavity flow problem:

- Understanding the effect of the time-step on ROM accuracy for predictive-in-time `SPARC` ROM simulations.
- Understanding the effect of various preconditioners (described in more detail in Objective 2) on ROM accuracy for predictive-in-time simulations.
- Understanding the effect of the training window and snapshot sampling frequency on ROM accuracy for predictive-in-time simulations.
- Determining the basis size required to achieve a viable predictive-in-time ROM.

"Viability" of the ROM is defined as the ability of the ROM to represent the pressure PSD of the flow at various points of interest (e.g., on the inner cavity wall; see Section 3.2). All findings will be documented in a SAND report and/or published article, and archived in a repository.

Towards ***Objective 2*** we plan to implement and understand the benefits of the following optimizations to the LSPG/POD ROM code in `SPARC`, required for achieving speedups over a high-fidelity `SPARC` simulation:

- Implementation of preconditioners in `SPARC` that emulate "ideal preconditioned ROMs" (Section 3.4.4). These preconditioners effectively change the inner product in which the residual norm is minimized, and are critical to obtaining ROMs that accurately reproduce the pressure PSDs for predictive-in-time simulations using the ROM. We plan to leverage the existing `SPARC` interface to `Trilinos` [20] preconditioners.
- Introduction of a QR algorithm for solving minimal residual least squares problem at the heart of the LSPG/POD Galerkin method for improved stability and faster ROM computa-

tions. We plan to leverage QR capabilities in `Trilinos`.

- Implementation of hyper-reduction via the "sample mesh" concept. In this approach, the residual in the ROM is computed only at a small set of points, rather than the entire mesh. The approach is necessary to achieve speed-ups with a ROM over its corresponding full order model.

Measurable outcomes will include a demonstration of the reduction in wall-clock time for a ROM run with each optimization performed, towards getting an idea of the speedups possible with LSPG/ROMs implemented within `SPARC`. We will also study numerically the effect of the planned optimizations on ROM accuracy.

Parallel FY18 development efforts within `SPARC` will likely include the implementation of physics-based structure-preserving constraints (Chapter 5 and [39]) as well as the implementation of a recently-proposed space-time MOR methodology which gives rise to ROMs with slower-growing error bounds [12].

References

[1] S. Arunajatesan, M. Bhardwaj, C. Wilson, and M. Ross. One-way coupled fluid structure simulations of stores in weapons bays. Proceedings of 51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition (AIAA 2013–0665), 2013.

[2] M. Balajewicz, I. Tezaur, and E. Dowell. Minimal subspace rotation on the stiefel manifold for stabilization and enhancement of projection-based reduced order models for the compressible navier-stokes equations. *J. Comput. Phys.*, 321:224–241, 2016.

[3] D. Barnette, R. Barrett, S. Hammond, J. Jayaraj, and J. Laros III. Using miniapplications in a mantevo framework for optimizing sandias sparc cfd code on multi-core, many-core, and gpu-accelerated compute platforms. Proceedings of 51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition (AIAA 2013–1126), 2013.

[4] M.F. Barone, I. Kalashnikova, D.J. Segalman, and H. Thornquist. Stable Galerkin reduced order models for linearized compressible flow. *J. Comput. Phys.*, 288:1932–1946, 2009.

[5] P. Bochev, D. Ridzal, and J. Young. Optimization-based modeling with applications to transport: Part 1. Abstract Formulation. *Large-Scale Scientific Computing (part of Lecture Notes in Computer Science book series)*, 7116:63–71, 2012.

[6] K. Carlberg. Adaptive *h*–refinement for reduced-order models. *arXiv e-Print*, 2014.

[7] K. Carlberg, M. Barone, and H. Antil. Galerkin v. least-squares Petrov–Galerkin projection in nonlinear model reduction. *J. Comput. Phys.*, 330:693–734, 2017.

[8] K. Carlberg, C. Farhat, J. Cortial, and D. Amsallem. The GNAT method for nonlinear model reduction: Effective implementation and application to computational fluid dynamics and turbulent flows. *J. Comput. Phys.*, 24(2):623–647, 2013.

[9] K. Carlberg, R. Tuminaro, and P. Boggs. Preserving Lagrangian structure in nonlinear model reduction with application to structural dynamics. *SIAM J. Sci. Comput.*, 2:B153–B184, 2015.

[10] M.H. Carpenter, T.C. Fisher, E.J. Nielsen, and S.H. Frankel. Entropy stable spectral collocation schemes for the Navier-Stokes equations: discontinuous interfaces. *SIAM J. Sci. Comput.*, 36:B835–B867, 2014.

[11] S. Chaturantabut and D.C. Sorensen. Nonlinear model reduction via discrete empirical interpolation. *SIAM J. Sci. Comput.*, 32:2737–2764, 2010.

[12] Y. Choi and K. Carlberg. Spacetime least-squares petrovgalerkin projection for nonlinear

model reduction. *arXiv e-Print*, (1703.04560), 2017.

[13] P. Constantin. On the euler equations of incompressible fluids. *Bulletin of the American Mathematical Society*, 44(4):603–621, 2007.

[14] R. Everson and L. Sirovich. Karhunen-Loeve procedure for gappy data. *J. Optical Society of America A*, pages 1657–1664, 1995.

[15] T.C. Fisher and M.H. Carpenter. High-order entropy stable finite difference schemes for nonlinear conservation laws: finite domains. *J. Comput. Phys.*, 252:518–557, 2013.

[16] M. Fujiwara. *Frontier of Physics in Fusion-Relevant Plasmas*. Proceedings of the Asian Science Seminar, Hefei, China, 1996.

[17] M. Gerritsen and P. Olsson. Designing an efficient solution strategy for fluid flows: 1. a stable high order finite different scheme and sharp shock resolution for the Euler equations. *J. Comput. Phys.*, 129:245–262, 1996.

[18] A. Harten. High resolution schemes for hyperbolic conservation laws. *Journal of Computational Physics*, 135(5):260–278, 1997.

[19] A. Harten and P.D. Lax. On a class of high resolution total-variation stable finite-difference schemes. *SIAM Journal on Numerical Analysis*, 21(1):1–23, 1984.

[20] M.A. Heroux, R.A. Bartlett, V.E. Howle, R.J. Hoekstra, J.J. Hu, T.G. Kolda, R.B. Lehoucq, K.R. Long, R.P. Pawlowski, E.T. Phipps, A.G. Salinger, H.K. Thornquist, R.S. Tuminaro, J.M. Willenbring, A. Williams, and K.S. Stanley. An overview of the Trilinos project. *ACM Trans. Math. Softw.*, 31(3), 2005.

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

[22] T.J.R. Hughes, L.P. Franca, and M. Mallet. A new finite element formulation for computational fluid dynamics: I. symmetric forms of the compressible Euler and Navier-Stokes equations and the second law of thermodynamics. *Comput. Meth. Appl. Mech. Engng.*, 54:223–234, 1986.

[23] A. Jameson. Origins and further development of the jameson-schmidt-turkel scheme. *AAIA Journal*, 55(5):1487–1510, 2017.

[24] I. Kalashnikova, S. Arunajatesan, M.F. Barone, B.G. van Bloemen Waanders, and J.A. Fike. Reduced order modeling for prediction and control of large-scale systems. Sandia National Laboratories Report, SAND No. 2014-4693. Sandia National Laboratories, Albuquerque, NM, 2014.

[25] I. Kalashnikova and M.F. Barone. Stable and efficient Galerkin reduced order models for non-linear fluid flow. AIAA-2011-3110, 6th AIAA Theoretical Fluid Mechanics Conference, Honolulu, HI, 2011.

[26] K. Kunisch and S. Volkwein. Galerkin proper orthogonal decomposition for a general equation in fluid dynamics. *SIAM J. Num. Anal.*, 40(2):492–515, 2002.

[27] P. LeGresley. Application of proper orthogonal decomposition (pod) to design decomposition methods. Ph.D. thesis, Stanford University, 2006.

[28] R.J. LeVeque. *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press, Cambridge, 2002.

[29] J.L. Lumley. *Stochastic tools in turbulence*. Academic Press: New York, 1971.

[30] D. Maddix, I. Tezaur, and K. Carlberg. Methods for structure preserving reduced order models in fluid dynamics. *In Preparation*, 2017.

[31] N.C. Nguyen, A.T. Patera, and J. Peraire. A 'best points' interpolation method for efficient approximation of parametrized functions. *Int. J. Numer. Meth. Engng.*, 73:521–543, 2008.

[32] N.C. Nguyen and J. Peraire. An efficient reduced-order modeling approach for non-linear parametrized partial differential equations. *Int. J. Numer. Meth. Engng.*, 76:27–55, 2008.

[33] A. Palha and M. Gerritsma. A mass, energy, enstrophy and vorticity conserving (meevc) mimetic spectral element discretization for the 2d incompressible navier-stokes equations. *Journal of Computational Physics*, 328:200–220, 2017.

[34] L. Peng and K. Carlberg. Structure-preserving model reduction for marginally stable LTI systems. *arXiv e-Print*, 1704.04009, 2017.

[35] M.M. Rai. Navier-Stokes simulations of blade-vortex interaction using high-order accurate upwind schemes. *AIAA Paper 87-0543*, 1987.

[36] M. Rathinam and L.R. Petzold. A new look at proper orthogonal decomposition. *SIAM J. Num. Anal.*, 41(5):1893–1925, 2003.

[37] L.G. Rebholz. Conservation laws of turbulence models. *J. Math. Anal. Appl.*, 326:33–45, 2007.

[38] P.L. Roe. Approximate Riemann solvers, parameter vectors and difference schemes. *J. Comput. Phys.*, 43:357–372, 1981.

[39] S. Sargsyan. Dimensionality hyper-reduction and machine learning for dynamical systems with varying parameters. Ph.D. Thesis, University of Washington, 2016.

[40] L. Sirovich. Turbulence and the dynamics of coherent structures, part iii: dynamics and scaling. *Q. Appl. Math.*, 45(3):583–590, 1987.

[41] G.A. Sod. A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws. *Journal of Computational Physics*, 27:1–31, 1978.

[42] J.L. Steger and R.F. Warming. Flux vector splitting of the inviscid gasdynamic equations

with application to finite-difference methods. *J. Comput. Phys.*, 40:263–293, 1981.

[43] P.K. Subbareddy and G.V. Candler. A fully discrete, kinetic energy consistent finite-volume scheme for compressible flows. *J. Comput. Phys.*, 228:1347–1364, 2009.

[44] Ringler T.D. and Randall D.A. A potential enstrophy and energy conserving numerical scheme for solution of the shallow-water equations on a geodesic grid. *Monthly Weather Review*, 130(5):1397–1409, 2002.

[45] I. Tezaur, J. Fike, K. Carlberg, M. Balajewicz, M. Barone, and E. Mussoni. Model reduction for compressible cavity simulations towards uncertainty quantification of structural loading. Sandia National Laboratories Report, Sand No. 2016-9463. Sandia National Laboratories, Albuquerque, NM, 2016.

[46] H.C. Yee, R.F. Warming, and A. Harten. *Implicit Total Variation Diminishing (TVD) Schemes for Steady-State Calculations*. *Journal of Computational Physics*, 57(3):327–260, 1985.

[47] M. Zhar. Mortestbed user guide, 2010.

DISTRIBUTION:

1	MS 0825	Srini Arunajatesan, 01515
1	MS 0825	Matt Barone, 01515
1	MS 0825	Steve Beresh, 01515
1	MS 1070	Jeff Fike, 01526
1	MS 9159	Kevin Carlberg, 08759
1	MS 9159	Payton Lindsay, 08759
1	MS 9957	Erin Mussoni, 08253
1	MS 9159	Jerry McNeish, 08754
1	MS 9159	Gayle Thayer, 08759
1	MS 9159	Liqian Peng, 08759
1	MS 0825	Micah Howard, 01515
1	MS 0845	Miguel Aguilo, 01542
1	MS 9159	Chi Hoang, 08759
1	MS 0899	Technical Library, 9536 (electronic copy)

Sandia National Laboratories