

SANDIA REPORT

SAND2016-9463

Unlimited Release

Printed September 2016

Model Reduction for Compressible Cavity Simulations Towards Uncertainty Quantification of Structural Loading

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

Prepared by

Sandia National Laboratories

Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

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>



Model Reduction for Compressible Cavity Simulations Towards Uncertainty Quantification of Structural Loading

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

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

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

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

Abstract

This report summarizes FY16 progress towards enabling uncertainty quantification for compressible cavity simulations using model order reduction (MOR). The targeted application is the quantification of the captive-carry environment for the design and qualification of nuclear weapons systems. To accurately simulate this scenario, Large Eddy Simulations (LES) require very fine meshes and long run times, which lead to *week*-long runs even on parallel state-of-the-art supercomputers. MOR can reduce substantially the CPU-time requirement for these simulations. We describe two approaches for model order reduction for nonlinear systems, which can yield significant speed-ups when combined with hyper-reduction: the Proper Orthogonal Decomposition (POD)/Galerkin approach and the POD/Least-Squares Petrov Galerkin (LSPG) approach. The implementation of these methods within the in-house compressible flow solver SPARC is discussed. Next, a method for stabilizing and enhancing low-dimensional reduced bases that was developed as a part of this project is detailed. This approach is based on a premise termed “minimal subspace rotation”, and has the advantage of yielding ROMs that are more stable and accurate for

long-time compressible cavity simulations. Numerical results for some laminar cavity problems aimed at gauging the viability of the proposed model reduction methodologies are presented and discussed.

Acknowledgment

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company for the United States Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Contents

1	Introduction	13
2	Nonlinear projection-based model order reduction	17
2.1	Reduced basis calculation via the Proper Orthogonal Decomposition	18
2.2	Projection	18
2.2.1	Galerkin ROMs	19
2.2.2	Least-Squares Petrov-Galerkin (LSPG) ROMs	19
2.3	Hyper-reduction	21
2.3.1	LSPG + Collocation ROM	22
2.3.2	GNAT method: LSPG + gappy POD	23
3	Implementation within the SPARC code	25
3.1	FOM capabilities in SPARC	25
3.2	ROM implementation in SPARC	25
3.2.1	Meshes	25
3.2.2	Time-integration	26
3.2.3	Projection	26
3.2.4	Linear solvers	26
3.2.5	Hyper-reduction	26
3.3	SPARC ROM utilities	27
3.3.1	POD basis generation routines	27
3.3.2	Sample mesh creation	28
3.3.3	Projection error	28
3.3.4	Post processing	28
4	Basis stabilization and enhancement via minimal subspace rotation	29
4.1	Application of basis rotation to Galerkin ROMs	30
4.1.1	Trace minimization problem for subspace rotation	31
4.1.1.1	Alternative objective functions and constraints	32
4.1.2	Solution of constrained optimization problem	33
4.1.3	Stabilization algorithm	34
4.2	Application of basis rotation to LSPG ROMs	36
5	Numerical results	37
5.1	Numerical evaluation of LSPG ROMs in SPARC	37
5.1.1	Viscous laminar cavity	37
	Reproductive ROMs	40
	Predictive ROM - Extended Time	47
	Predictive FOM - Mach Number Variation	49

5.1.2	Inviscid pulse in uniform base flow	49
5.2	Numerical evaluation of ROM stabilized via minimal subspace rotation	54
5.2.1	Sigma CFD–Spirit Galerkin ROM stabilized via minimal subspace rotation for the viscous laminar cavity	55
5.2.2	SPARC–Spirit LSPG ROM stabilized via minimal subspace rotation for the inviscid pulse in uniform base flow	58
6	Discussion	61
6.1	LSPG ROMs in SPARC	61
6.2	Basis stabilization and enhancement via minimal subspace rotation	62
6.3	FY17 project plan	62
	References	63

List of Figures

1.1	Compressible captive-carry problem	13
5.1	The computational mesh used for 2D viscous laminar cavity simulations in SPARC.	38
5.2	The pressure time history for a point midway up the downstream wall of the cavity.	40
5.3	The pressure time histories at a point midway up the downstream wall of the cavity for the FOM and reproductive ROMs using a POD basis constructed from 800 snapshots.	41
5.4	The discarded energy fractions for the POD basis constructed from 800 snapshots for the cavity problem.	42
5.5	The projection error for the POD basis constructed from 800 snapshots for the cavity problem. The dashed line indicates the end of the snapshot collection period.	43
5.6	The pressure time histories at a point midway up the downstream wall of the cavity for the FOM and reproductive ROMs using a POD basis constructed from 2000 snapshots.	44
5.7	The discarded energy fractions for the POD basis constructed from 2000 snapshots for the cavity problem.	45
5.8	The projection error for the POD basis constructed from 2000 snapshots for the cavity problem.	46
5.9	The power spectral density of the pressure fluctuations measured at a point midway up the downstream wall of the cavity.	46
5.10	The pressure time histories at a point midway up the downstream wall of the cavity for the FOM and reproductive ROMs using a POD basis constructed from 2000 snapshots when the tolerances are varied.	47
5.11	The pressure time history for a point midway up the downstream wall of the cavity when an updated version of SPARC is run with tighter tolerances and for a longer period of time. The pressure does not appear to reach a periodic state.	48
5.12	The pressure time histories at a point midway up the downstream wall of the cavity for the FOM and ROMs run for an extended time period using a POD basis constructed from 800 snapshots. The dashed vertical line indicates the end of the snapshot collection period. Curves terminating before time 0.154 indicate that a non-physical quantity was computed by the ROM, and the ROM was unable to continue.	50
5.13	The pressure time histories at a point midway up the downstream wall of the cavity for the FOM and ROMs run at Mach 0.6 when the POD basis was constructed using snapshots from Mach 0.575 and Mach 0.625 runs. Curves terminating before time 0.02 indicate that a non-physical quantity was computed by the ROM, and the ROM was unable to continue.	51

5.14	The projection errors for the POD basis constructed using the FOM run dimensionally (left) vs. non-dimensionally (right) for the inviscid pulse problem. The POD basis for each case was constructed using every 5 snapshots, but the projection error is computed for every snapshot. This can cause the error to increase substantially for snapshots not in the training set	53
5.15	Time history of SPARC LSPG ROM generalized coordinates for mode 4 compared to FOM projection for the inviscid pulse problem run dimensionally (left) vs. non-dimensionally (right).	54
5.16	Nonlinear model reduction of the cavity at $Re \approx 5500$. Evolution of modal energy; DNS (thick gray line), standard $M = 20$ ROM (dashed blue line), stabilized $M = P = 20$ ROM (solid black line).	56
5.17	PSD of pressure at the point half way up the downstream wall of the cavity for the cavity at $Re \approx 5500$. DNS (thick gray line), stabilized $M = P = 20$ ROM (black line)	56
5.18	CPSD of the pressure between two points inside the of the cavity at $Re \approx 5500$. DNS (thick gray line), stabilized $M = P = 20$ ROM (black line)	57
5.19	Snapshot of the cavity $Re \approx 5500$; contours of u -velocity magnitude at the final snapshot. DNS (top), standard $M = 20$ ROM (middle), and stabilized $M = P = 20$ ROM (bottom)	58
5.20	Time history of SPARC-Spirit $M = 10$ mode Galerkin ROM generalized coordinates compared to FOM projection for inviscid pulse problem run: marginal stability constraint.	60
5.21	Time history of SPARC-Spirit $M = 10$ mode Galerkin ROM generalized coordinates compared to FOM projection for inviscid pulse problem run: asymptotic stability constraint.	60

List of Tables

5.1	Parameters used for the 2D viscous laminar cavity test case in SPARC.....	39
5.2	Parameters used for the inviscid pulse test case.	52

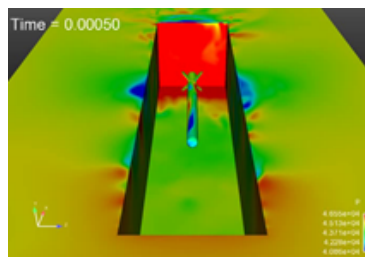
Chapter 1

Introduction

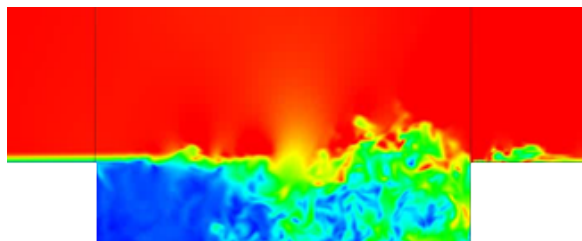
The quantification of the captive-carry environment is of great interest to Sandia National Laboratories, as it enables the design and qualification of nuclear weapons (NW) systems and related components. In the captive-carry problem, a weapons bay (Figure 1.1) and its contents experience large unsteady pressure loads (which can exceed 160 dB for certain bay geometries and flow conditions) when exposed to an external flow field. Special care must be taken to design these components such that they are able to withstand loads of this magnitude. There exist in-house Large Eddy Simulation (LES) Computational Fluid Dynamics (CFD) codes that can be used to simulate this scenario, e.g., the Sigma CFD [1] and SPARC [7] flow solvers. In order to predict the dynamic loads on the cavity and its components to a sufficient accuracy very fine meshes and long run times are required. Despite improved algorithms and more powerful computing platforms, a single relevant captive-carry simulation can take on the order of *weeks* to complete even when run on parallel, state-of-the-art supercomputers. These computational requirements preclude advanced analysis of the captive-carry environment even with the projected growth in computing power. In particular, uncertainty quantification (UQ) is intractable, as it requires running a simulation numerous times.



(a) Weapons bay of an airplane



(b) Compressible cavity with object



(c) 2D cross-section of compressible cavity

Figure 1.1. Compressible captive-carry problem

A promising tool for drastically reducing compute times and enabling on-the-spot analysis and UQ is model order reduction (MOR). A reduced order model (ROM) is a model constructed from a high-fidelity simulation that retains the essential physics and dynamics of its corresponding full order model (FOM), but has a much smaller computational cost. Although some consider a data-fit or low-fidelity model a reduced order model, herein the term “reduced order model” refers to a projection-based ROM. In projection-based MOR, the state variables are approximated by a low-dimensional subspace, typically obtained through a data compression performed on a set of snapshots collected from a high-fidelity simulation or physical experiment, followed by truncation: the removal of modes believed to be unimportant in representing a problem solution. There are numerous approaches in the literature for computing a low-dimensional subspace, e.g., Proper Orthogonal Decomposition (POD) [42, 20], Dynamic Mode Decomposition (DMD) [37, 41] balanced POD (BPOD) [36, 46], balanced truncation [18, 29], and the reduced basis method [39, 44]. Once a reduced basis is computed, the ROM dynamical system is obtained by projecting the governing equations, or some discretized form of these equations, onto the low-dimensional subspace. For non-linear problems, an additional layer of approximation, referred to as “hyper-reduction”, is usually required to gain a computational speed-up.

This report summarizes an ongoing research effort to develop stable, accurate, robust and efficient projection-based model reduction methodologies for compressible flow problems, and to implement the most viable of these approaches in the SPARC in-house flow solver. The effort leverages earlier work by various team-members, most notably, the following

- Barone, Kalashnikova, Fike *et al.* [9, 23, 22]: It is demonstrated that POD/Galerkin ROMs constructed in the L^2 inner product for compressible flows can possess non-physical instabilities. A stable formulation can be obtained through the careful construction of a weighted L^2 inner product, derived using energy methods, for both linear and nonlinear compressible flow equations.
- Balajewicz, Tezaur *et al.* [5, 4]: The authors argue that, for a projection-based ROM of a fluid flow to be stable and accurate, the dynamics of the truncated subspace must be taken into account. An approach for stabilizing and enhancing projection-based fluid ROMs in which truncated modes are accounted for *a priori* via a minimal rotation of the projection subspace is proposed.
- Carlberg *et al.* [14]: The Gauss-Newton with Approximated Tensors (GNAT) method for nonlinear model reduction, a Least Squares Petrov-Galerkin (LSPG) minimal-residual-based approach that creates the ROM at the fully discrete (rather than semi-discrete) level, and maintains efficiency using gappy POD [16] is proposed. This approach is proven to be discrete-optimal, and shown numerically to have better stability properties than the POD/Galerkin method for compressible flow problems.

In order to be considered viable, a ROM must be capable of capturing the statistics (e.g., pressure power spectral density, or PSD) of a flow for long-time simulations to a sufficient accuracy. Of particular importance are time-predictive simulations using the ROM; that is, predictability for simulations in which the ROM is run at the same parameter values as the FOM from which it was constructed, but for much longer in time. While the situation in which the ROM is run at different

parameter values (e.g., Reynolds and/or Mach number) than the FOM is of interest as well for enabling UQ, ROM robustness with respect to parameter changes is less critical than predictability in time.

A difficulty encountered by projection-based ROMs for compressible flow problems is ROM instability: as demonstrated in [9, 22, 11], a compressible fluid ROM might be stable using a given number of modes, but unstable for other choices of basis size. These non-physical ROM instabilities can limit the usefulness of ROMs for long-time predictions. Since it was shown in earlier work by Carlberg, Barone *et al.* [14, 12] that LSPG ROMs, which are based on Petrov-Galerkin projection, tend to be more stable than Galerkin ROMs, we chose to implement this flavor of methods in the MOR suite within SPARC, in addition to the traditional Galerkin projection method (for comparison purposes). A parallel effort to the implementation and testing of these MOR methods in SPARC has been the extension of the method of minimal subspace rotation [5, 4], a non-intrusive basis stabilization method, to compressible flow problems simulated in SPARC. This approach has been demonstrated to produce ROMs that are predictive in time (and hence amenable to long-time simulations) for laminar compressible cavity problems [4]. Moreover, the method of minimal subspace rotation can enable extreme model reduction without sacrificing accuracy. Basis stabilization methods were deemed necessary for LSPG ROMs because, while more stable than Galerkin ROMs, LSPG ROMs *can* exhibit instabilities for low-dimensional expansions.

Toward this effect, the remainder of this report is organized as follows. Projection-based model order reduction (the Galerkin method, the LSPG method and gappy POD for hyper-reduction) for general nonlinear systems is overviewed in Chapter 2. The implementation of model reduction capabilities within the SPARC flow solver is detailed in Chapter 3. Next, Chapter 4 motivates and describes reduced basis stabilization and enhancement via minimal subspace rotation. Finally, numerical results are presented in Chapter 5 and discussed in Chapter 6. These results provide some insight into the viability of the considered model reduction methodologies for compressible captive-carry problems.

Chapter 2

Nonlinear projection-based model order reduction

In this chapter, we overview some classical projection-based approaches to model reduction for nonlinear partial differential equations (PDEs). There are three classes of projection-based model reduction approaches:

- (i) approaches which operate directly on the governing PDEs at the continuous level [9, 22],
- (ii) approaches which operate on a semi-discrete form of the governing PDEs (i.e., the PDEs discretized in space, but not in time), and
- (iii) approaches which operate on the fully discretized governing PDEs (i.e., the PDEs discretized in space and time) [14].

Here, we restrict our attention to the third class of approaches, as it is this class of methods that was implemented within the SPARC flow solver.

Consider the following system of nonlinear equations

$$\mathbf{r}(\mathbf{w}) = \mathbf{0} \quad (2.1)$$

where $\mathbf{w} \in \mathbb{R}^N$ is the state vector and $\mathbf{r} : \mathbb{R}^N \rightarrow \mathbb{R}^N$ is the nonlinear residual operator. In our 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 we solve (2.1) using a (globalized) Newton's method, the sequence of solutions we generate are

$$\mathbf{J}^{(k)} \delta \mathbf{w}^{(k)} = -\mathbf{r}^{(k)}, \quad k = 1, \dots, K \quad (2.2)$$

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

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

Our task is to build a ROM for (2.1) using the projection-based model reduction approach. This approach consists of three steps:

- (i) calculation of a reduced basis,

- (ii) projection of the governing equations (in our case, (2.1)) onto the subspace spanned by the reduced basis, and
- (iii) hyper-reduction to handle efficiently the projection of the nonlinear terms.

Each of these steps is outlined below. The numerical implementation of the model reduction algorithms outlined in this chapter within the SPARC flow solver is described later, in Chapter 3.

2.1 Reduced basis calculation via the Proper Orthogonal Decomposition

The first step of any projection-based model reduction approach is the calculation of a reduced basis of order M with $M \ll N$, denoted by $\Phi_M = (\phi_1, \dots, \phi_M)$. There are numerous approaches for calculating the reduced basis modes¹. Here, we restrict our attention to reduced bases computed via the Proper Orthogonal Decomposition (POD) [42, 3, 20]. 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 $\{\mathbf{x}^k : k = 1, \dots, K\}$ 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 ($M \ll K$) subspace spanned by the set $\{\phi_i\}$ 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 [9, 20, 26, 34] that the solution to the POD optimization problem reduces to the eigenvalue problem

$$\mathbf{R}\phi = \lambda\phi, \quad (2.4)$$

where \mathbf{R} is a self-adjoint and positive semi-definite operator with its (i, j) entry given by $R_{ij} = \frac{1}{K} (\mathbf{x}^i, \mathbf{x}^j)$ for $1 \leq i, j \leq K$. It can be shown [20, 28] that the set of M eigenfunctions, or POD modes, $\{\phi_i : i = 1, \dots, M\}$ 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 [42].

2.2 Projection

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

$$\tilde{\mathbf{w}} = \bar{\mathbf{w}} + \Phi_M \hat{\mathbf{w}} = \bar{\mathbf{w}} + \sum_{i=1}^M \phi_i \hat{w}_i \quad (2.5)$$

¹Approaches for calculating reduced bases include: the POD method [42, 3, 20], the BPOD method [46, 36], the balanced truncation method [29, 18], and the reduced basis method [44, 39]. There are also methods based on goal-oriented bases [11], generalized eigenmodes [6], and Koopman modes [38].

with $\hat{\mathbf{w}} := [\hat{w}_1 \cdots \hat{w}_M]^T \in \mathbb{R}^M$ denoting the generalized coordinates, and $\bar{\mathbf{w}} \in \mathbb{R}^N$ 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

$$\mathbf{r}(\bar{\mathbf{w}} + \Phi_M \hat{\mathbf{w}}) = \mathbf{0}, \quad (2.6)$$

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

Here, we consider two ways to compute a unique solution $\hat{\mathbf{w}}$: Galerkin projection and Least-Squares Petrov-Galerkin (LSPG) projection.

2.2.1 Galerkin ROMs

Performing a classical Galerkin projection of (2.6) onto the reduced basis modes in the discrete L^2 inner product, we obtain the following system

$$\Phi_M^T \mathbf{r}(\bar{\mathbf{w}} + \Phi_M \hat{\mathbf{w}}_G) = 0, \quad (2.7)$$

where the approximate solution is $\tilde{\mathbf{w}}_G := \bar{\mathbf{w}} + \Phi_M \hat{\mathbf{w}}_G$.

Despite its popularity, the Galerkin approach to model reduction is known to have some disadvantages when applied to compressible flow problems. In particular, it is well-known that reduced order models constructed using the POD/Galerkin method lack in general an *a priori* stability guarantee. Thus, it can happen that a POD/Galerkin constructed for a stable full order model exhibits a non-physical instability, which can lead to practical limitations of the ROM: as demonstrated in [11, 24, 25, 23, 9], a compressible fluid POD/Galerkin ROM might be stable for a given number of modes, but unstable for other choices of basis size.

2.2.2 Least-Squares Petrov-Galerkin (LSPG) ROMs

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

$$\hat{\mathbf{w}}_{PG} = \arg \min_{\mathbf{y} \in \mathbb{R}^M} \|\mathbf{r}(\bar{\mathbf{w}} + \Phi_M \mathbf{y})\|_2^2. \quad (2.8)$$

Here, the approximate solution is $\tilde{\mathbf{w}}_{PG} := \bar{\mathbf{w}} + \Phi_M \hat{\mathbf{w}}_{PG}$. The name ‘‘LSPG’’ ROM comes from the observation that solving (2.8) 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.* [14], we adopt the Gauss–Newton

approach². This approach implies solving a sequence of linear least-squares problems of the form

$$\delta \hat{\mathbf{w}}_{\text{PG}}^{(k)} = \arg \min_{\mathbf{y} \in \mathbb{R}^M} \|\mathbf{J}^{(k)} \Phi_M \mathbf{y} + \mathbf{r}^{(k)}\|_2^2, \quad k = 1, \dots, K_{\text{PG}} \quad (2.9)$$

$$\hat{\mathbf{w}}_{\text{PG}}^{(k)} = \hat{\mathbf{w}}_{\text{PG}}^{(k-1)} + \alpha_k \delta \hat{\mathbf{w}}_{\text{PG}}^{(k)} \quad (2.10)$$

$$\tilde{\mathbf{w}}_{\text{PG}}^{(k)} = \tilde{\mathbf{w}} + \Phi_M \hat{\mathbf{w}}_{\text{PG}}^{(k-1)} \quad (2.11)$$

It can be shown that the approximation upon convergence is $\tilde{\mathbf{w}}_{\text{PG}} = \tilde{\mathbf{w}}_{\text{PG}}^{(K_{\text{PG}})}$ and $\hat{\mathbf{w}}_{\text{PG}} = \hat{\mathbf{w}}_{\text{PG}}^{(K)}$.³ Note that the normal-equations form of (2.9) is

$$\Phi_M^T \mathbf{J}^{(k)T} \mathbf{J}^{(k)} \Phi_M \delta \hat{\mathbf{w}}_{\text{PG}}^{(k)} = -\Phi_M^T \mathbf{J}^{(k)T} \mathbf{r}^{(k)}, \quad k = 1, \dots, K_{\text{PG}}, \quad (2.12)$$

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

The simplest implementation of the Gauss–Newton method for solving (2.8) consists of the following steps:

- (i) Compute the residual $\mathbf{r}^{(k)}$ and Jacobian $\mathbf{J}^{(k)}$.
- (ii) Compute the product $\mathbf{J}^{(k)} \Phi_M$.
- (iii) Compute the low-dimensional Jacobian $\left[\mathbf{J}^{(k)} \Phi_M \right]^T \left[\mathbf{J}^{(k)} \Phi_M \right]$.
- (iv) Compute the low-dimensional residual $\left[\mathbf{J}^{(k)} \Phi_M \right]^T \mathbf{r}^{(k)}$.
- (v) Solve linear system (2.12).
- (vi) Perform updates (2.10)–(2.11).
- (vii) Set $k \leftarrow k + 1$ and return to step 1.

Once this simple implementation is complete and has been verified, step (v) should be replaced with a more numerically stable linear least-squares (e.g., from ScaLAPACK) solve for (2.9). This would involve computing a QR or singular value decomposition (SVD) factorization.

As shown in [14], LSPG ROM approaches, e.g., the GNAT method of Carlberg *et al.*, tend to possess better stability properties than ROMs constructed via Galerkin projection. In particular, unpublished results showed that the GNAT ROMs implemented within the AERO-F flow solver⁴ were capable of generating stable predictions beyond the time interval in which it was trained. In addition, recent thorough investigations carried out by the team at Sandia [12] illustrated the following relationships between LSPG and Galerkin ROMs:

²The LSPG approach is the basis for the Gauss–Newton with Approximated Tensors (GNAT) method of Carlberg *et al.* [14].

³In the event of an unsteady simulation, the initial guess for the generalized coordinates is taken to be the generalized coordinates at the previous timestep.

⁴AERO-F is available for download at <https://bitbucket.org/frg/aero-f>.

(i) Continuous and discrete representations

- Galerkin projection and time discretization are commutative.
- LSPG ROMs can be derived for Runge–Kutta schemes.
- The LSPG ROM has a time-continuous (i.e., ordinary differential equation, or ODE) representation under certain conditions. This ODE depends on the time step.

(ii) Equivalence conditions

- Galerkin and LSPG ROMs are equivalent for explicit time integrators.
- Galerkin and LSPG ROMs are equivalent in the limit of $\Delta t \rightarrow 0$ for implicit time integrators.
- Galerkin ROMs are discrete optimal for symmetric-positive-definite residual Jacobians. Thus, in this case, LSPG and Galerkin ROMs are equivalent for a specific choice of the weighted ℓ^2 norm employed by LSPG.

(iii) Error analysis

- For backward differentiation formulas, the LSPG ROM can yield a lower local *a posteriori* error bound than the Galerkin ROM because it solves a time-global optimization problem (over a time window) rather than a time-local optimization problem .
- For the backward Euler time integrator, *an intermediate* time step should yield the lowest error bound.
- For the backward Euler time integrator, a larger basis size leads to a smaller optimal time step for the LSPG ROM.

This study also demonstrated the promise of the GNAT ROM for the application at hand: it yielded accurate (sub-3% errors) responses for a compressible cavity flow, with computational savings (measured in core-hours) exceeding $40\times$.

2.3 Hyper-reduction

Unfortunately, neither the Galerkin nor the LSPG projection approach detailed 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) [15], “best points” interpolation [30, 31], collocation [27] and gappy POD [16]. Here, we detail two of these hyper-reduction algorithms, namely collocation and gappy POD. The implementation of these methods

is described in the context of the LSPG model reduction approach. LSPG combined with gappy POD is equivalent to the GNAT method [14].

Note that introducing hyper-reduction produces a less accurate model *in principle*, as an additional approximation has been employed. This leads to the notion of a hierarchy of models [13]:

- I. Full-order model
- II. Projection-based reduced-order model (e.g., Galerkin, LSPG)
- III. Projection-based reduced-order model with hyper-reduction (e.g., Galerkin–DEIM, GNAT = LSPG + gappy POD)

In the context of LSPG projection specifically, hyper-reduction has the effect of changing the norm in which the residual is minimized. This effectively alters the norm in which the residual is minimized by the ROM (compare (2.8) and (2.18)). Thus, it is commonly observed *in practice* that introducing hyper-reduction does not significantly hamper the quality of the LSPG ROM.

2.3.1 LSPG + Collocation ROM

The key idea behind collocation is to compute only a small number of rows of the residual vector and Jacobian matrix online. Assume we have a ‘sampling matrix’ $\mathbf{Z} \in \{0, 1\}^{q \times N}$ with $M \leq q \ll N$ that consists of selected rows of the identity matrix. This matrix can be obtained, e.g. via a greedy algorithm, and dictates which rows of the residual and Jacobian we will compute online. Given \mathbf{Z} , one can mitigate the computational burden described at the end of Section 2.2.2 by replacing $\mathbf{r}^{(k)}$ with $\mathbf{Z}\mathbf{r}^{(k)}$ and $\mathbf{J}^{(k)}$ with $\mathbf{Z}\mathbf{J}^{(k)}$. In this case, the nonlinear least-squares problem (2.8) becomes

$$\hat{\mathbf{w}}_{\text{coll,PG}} = \arg \min_{\mathbf{y} \in \mathbb{R}^M} \|\mathbf{Z}\mathbf{r}(\bar{\mathbf{w}} + \Phi_M \mathbf{y})\|_2^2, \quad (2.13)$$

and the associated Gauss–Newton iterations are

$$\delta \hat{\mathbf{w}}_{\text{coll,PG}}^{(k)} = \arg \min_{\mathbf{y} \in \mathbb{R}^M} \|\mathbf{Z}\mathbf{J}^{(k)} \Phi_M \mathbf{y} + \mathbf{Z}\mathbf{r}^{(k)}\|_2^2, \quad k = 1, \dots, K_{\text{coll,PG}} \quad (2.14)$$

$$\hat{\mathbf{w}}_{\text{coll,PG}}^{(k)} = \hat{\mathbf{w}}_{\text{coll,PG}}^{(k-1)} + \alpha_k \delta \hat{\mathbf{w}}_{\text{coll,PG}}^{(k)} \quad (2.15)$$

$$\tilde{\mathbf{w}}_{\text{coll,PG}}^{(k)} = \bar{\mathbf{w}} + \Phi_M \hat{\mathbf{w}}_{\text{coll,PG}}^{(k-1)} \quad (2.16)$$

Then, the approximation upon convergence is $\tilde{\mathbf{w}}_{\text{coll,PG}} = \tilde{\mathbf{w}}_{\text{coll,PG}}^{(K_{\text{coll,PG}})}$ and $\hat{\mathbf{w}}_{\text{coll,PG}} = \hat{\mathbf{w}}_{\text{coll,PG}}^{(K)}$. As before, the relevant nonlinear least-squares problem can be solved via the normal equations. The normal equations counterpart to (2.14) is

$$\Phi_M^T \mathbf{J}^{(k)T} \mathbf{Z}^T \mathbf{Z} \mathbf{J}^{(k)} \Phi_M \delta \hat{\mathbf{w}}_{\text{PG}}^{(k)} = -\Phi_M^T \mathbf{J}^{(k)T} \mathbf{Z}^T \mathbf{Z} \mathbf{r}^{(k)}, \quad k = 1, \dots, K_{\text{coll,PG}}, \quad (2.17)$$

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

As for the LSPG method without collocation, we outline the simplest algorithm for solving (2.13):

- (i) Compute only q rows of the residual $\mathbf{Z}\mathbf{r}^{(k)}$ and Jacobian $\mathbf{Z}\mathbf{J}^{(k)}$.
- (ii) Compute the product $\left[\mathbf{Z}\mathbf{J}^{(k)}\right]\Phi_M$, which is a $q \times M$ matrix.
- (iii) Compute the low-dimensional Jacobian $\left[\mathbf{Z}\mathbf{J}^{(k)}\Phi_M\right]^T \left[\mathbf{Z}\mathbf{J}^{(k)}\Phi_M\right]$.
- (iv) Compute the low-dimensional residual $\left[\mathbf{Z}\mathbf{J}^{(k)}\Phi_M\right]^T \mathbf{Z}\mathbf{r}^{(k)}$.
- (v) Solve linear system (2.17).
- (vi) Perform updates (2.15)–(2.16).
- (vii) Set $k \leftarrow k + 1$ and return to step 1.

As before, once this initial simple implementation is complete and has been verified, step (v) should be replaced with a more numerically stable linear least-squares solve for (2.14).

2.3.2 GNAT method: LSPG + gappy POD

The GNAT method [14] is similar to the LSPG + collocation approach described in Section 2.3.1. The primary difference is that the gappy POD approximation aims to approximate the entire residual and Jacobian (via least-squares approximation) rather than simply subsample those quantities. In particular, $\mathbf{r}^{(k)}$ is replaced with $\mathbf{W}(\mathbf{Z}\mathbf{W})^+\mathbf{Z}\mathbf{r}^{(k)}$ and $\mathbf{J}^{(k)}$ with $\mathbf{W}(\mathbf{Z}\mathbf{W})^+\mathbf{Z}\mathbf{J}^{(k)}$, where $\mathbf{W} \in \mathbb{R}^{N \times r}$ with $M \leq r \leq q \ll N$ is an orthogonal (i.e., $\mathbf{W}^T\mathbf{W} = \mathbf{I}$) reduced basis for the residual.⁵ Then, the method solves the nonlinear least-squares problem

$$\hat{\mathbf{w}}_{\text{GNAT}} = \arg \min_{\mathbf{y} \in \mathbb{R}^M} \|(\mathbf{Z}\mathbf{W})^+\mathbf{Z}\mathbf{r}(\tilde{\mathbf{w}} + \Phi_M\mathbf{y})\|_2^2 \quad (2.18)$$

via the Gauss–Newton iterations

$$\delta \hat{\mathbf{w}}_{\text{GNAT}}^{(k)} = \arg \min_{\mathbf{y} \in \mathbb{R}^M} \|(\mathbf{Z}\mathbf{W})^+\mathbf{Z} \left[\mathbf{J}^{(k)}\Phi_M\mathbf{y} + \mathbf{r}^{(k)} \right]\|_2^2, \quad k = 1, \dots, K_{\text{GNAT}} \quad (2.19)$$

$$\hat{\mathbf{w}}_{\text{GNAT}}^{(k)} = \hat{\mathbf{w}}_{\text{GNAT}}^{(k-1)} + \alpha_k \delta \hat{\mathbf{w}}_{\text{GNAT}}^{(k)} \quad (2.20)$$

$$\tilde{\mathbf{w}}_{\text{GNAT}}^{(k)} = \tilde{\mathbf{w}} + \Phi_M \hat{\mathbf{w}}_{\text{GNAT}}^{(k-1)} \quad (2.21)$$

One can show that the approximation upon convergence is $\tilde{\mathbf{w}}_{\text{GNAT}} = \tilde{\mathbf{w}}_{\text{GNAT}}^{(K_{\text{GNAT}})}$ and $\hat{\mathbf{w}}_{\text{GNAT}} = \hat{\mathbf{w}}_{\text{GNAT}}^{(K)}$. Again, the normal-equations counterpart to (2.19) is

$$\overline{\mathbf{J}^{(k)}}^T \overline{\mathbf{J}^{(k)}} \delta \hat{\mathbf{w}}_{\text{PG}}^{(k)} = -\overline{\mathbf{J}^{(k)}}^T \overline{\mathbf{r}^{(k)}}, \quad k = 1, \dots, K_{\text{GNAT}}, \quad (2.22)$$

where $\overline{\mathbf{J}^{(k)}} := (\mathbf{Z}\mathbf{W})^+ \left[\mathbf{Z}\mathbf{J}^{(k)} \right] \Phi_M \in \mathbb{R}^{r \times M}$ and $\overline{\mathbf{r}^{(k)}} := (\mathbf{Z}\mathbf{W})^+ \left[\mathbf{Z}\mathbf{r}^{(k)} \right] \in \mathbb{R}^r$ which can be interpreted as a Petrov–Galerkin process of the Newton iteration with trial basis (in matrix form) Φ_M and test basis $\mathbf{Z}^T \mathbf{Z} \mathbf{J}^{(k)} \Phi_M$.

⁵In practice, we should consider constructing a separate reduced basis for each component of the residual.

A basic implementation of the GNAT method is as follows:

- (i) Compute only q rows of the residual $\mathbf{Z}\mathbf{r}^{(k)}$ and Jacobian $\mathbf{Z}\mathbf{J}^{(k)}$.
- (ii) Compute the products $\overline{\mathbf{J}^{(k)}}$ and $\overline{\mathbf{r}^{(k)}}$.
- (iii) Compute the low-dimensional Jacobian $\overline{\mathbf{J}^{(k)}}^T \overline{\mathbf{J}^{(k)}}$.
- (iv) Compute the low-dimensional residual $\overline{\mathbf{J}^{(k)}}^T \overline{\mathbf{r}^{(k)}}$.
- (v) Solve linear system (2.22).
- (vi) Perform updates (2.20)–(2.21).
- (vii) Set $k \leftarrow k + 1$ and return to step 1.

Once the steps above have been implemented and this simple code verified, step (v) should be replaced with a more numerically stable linear least-squares solve as in (2.19).

Chapter 3

Implementation within the SPARC code

Reduced-order modeling capabilities have been added to the computational fluid dynamics code SPARC [7]. These capabilities include the reduced-order modeling implementation in SPARC itself and several supporting utilities.

3.1 FOM capabilities in SPARC

SPARC is a cell-centered, finite-volume code written in C++ that discretizes the compressible Navier-Stokes equations in conservative, dimensional form. Its salient features at the present time include:

- support for multiblock structured meshes, unstructured meshes and hybrid meshes,
- standard 2nd-order flux scheme¹,
- native solvers and vector/matrix classes,
- optional access to Trilinos [19] linear solver and vector/matrix data structures,
- explicit Runge-Kutta and implicit BDF time-marching schemes,
- I/O using CGNS files².

3.2 ROM implementation in SPARC

3.2.1 Meshes

Although SPARC supports multiblock structured meshes, unstructured meshes and hybrid meshes, it was decided that the ROM implementation would initially target only structured meshes, as this

¹The addition of higher-order schemes (e.g., the Rai scheme) is planned for FY17.

²See <http://cgns.github.io> for more on the CGNS file format.

should provide the capabilities desired for our target problem of interest.

3.2.2 Time-integration

The ROM implementation in `SPARC` consists primarily of modified versions of the time solvers. Specialized ROM child classes have been defined that allow `SPARC` to be run in either FOM mode or ROM mode simply by changing the time solver specified in the input file.

As mentioned in Section 3.1, `SPARC` contains explicit Runge-Kutta as well as implicit BDF schemes. The ROM implementation for the implicit schemes follows the approach summarized in Section 2.2.2. A similar approach was used for the ROM implementation in the specific case of an explicit time integration scheme.

3.2.3 Projection

The default implementation in `SPARC` constructs an LSPG ROM. This approach requires that an implicit time-integration scheme be used. The option to create Galerkin ROMs (for implicit as well as explicit time-integration schemes) is also available.

3.2.4 Linear solvers

Although `SPARC` has native matrix/vector data structures and linear solvers, the ROM implementation in `SPARC` uses the `Tpetra` vector and matrix classes from `Trilinos`. The reduced ROM system given in equation (2.12) is solved using the direct solvers from the `Trilinos` package `Amesos2`.

3.2.5 Hyper-reduction

As discussed in Section 2.3, the implementation described above does not provide a reduction in computational cost. This is because the residual and Jacobian are evaluated for every cell, as in the full-order model implementation. In addition to this, there are several operations involving full-scale matrix/vector multiplications. Although the solution of the reduced ROM system is much cheaper than the solution of the full order system, the overall computation time is dominated by the matrix and vector operations, especially when a large number of modes is used. In order to reduce the computational cost, a hyper-reduction technique is required.

In our implementation of hyper-reduction, we use the concept of a sample mesh [13] to define a small set of cells where the residual and Jacobian are evaluated. Since we run `SPARC` using its structured mesh implementation, a lot of computational efficiency comes through an efficient

traversal of the mesh. This efficient traversal is no longer possible with the introduction of a sample mesh. As a consequence, the use of a sample mesh is much more challenging than it would be in an unstructured mesh implementation.

The current sample mesh implementation is in a `SPARC` child class that redefines the residual and Jacobian calculations. This implementation works by creating a list of cells where the residual and Jacobian entries are evaluated. `SPARC` can switch between the nominal traversal through the mesh and this list traversal by specifying the appropriate discretization type in the input file. Either option can be used for the FOM or the ROM implementation, although the list traversal approach is currently limited to explicit schemes only.

The list implementation of the sample mesh described above is not the most optimal approach. Improvements to the sample mesh implementation were begun but put on hold until further testing and debugging of the ROM implementation is completed.

3.3 SPARC ROM utilities

3.3.1 POD basis generation routines

These routines read in a CGNS file containing snapshots of the flow field, compute a POD basis, and write the basis to a new CGNS file that can be read in by the `SPARC ROM` implementation. The POD basis calculation is performed using the `RBGen` library from the Trilinos package `Anasazi`. Several options specify how the basis is constructed:

- *Base flow type* specifies a quantity that is subtracted from the snapshots prior to basis generation. This would be the quantity $\bar{\mathbf{w}}$ in equation (2.5). The current options are:
 - *None* - nothing is subtracted.
 - *Mean* - the average flow field is computed and subtracted from all the snapshots.
 - *First* - the first snapshot is subtracted from all the other snapshots (default).
 - *Previous* - each snapshot has the previous snapshot subtracted from it so that the basis is computed using the increment between snapshots.
- *Basis Type*
 - *Vector* - a single basis is constructed for all flow variables (default).
 - *Scalar* - separate bases are constructed for each individual flow variable.
- *Non-Dimensionalization*
 - Specifies scaling factors that are applied to each variable prior to computing the basis

Note that the projection step of the model reduction is applied to the equations in *dimensional* variables.

3.3.2 Sample mesh creation

This routine creates a list of cells where the residual and Jacobian are to be evaluated. The algorithm used is adapted from the Model-Order Reduction (MOR) capabilities implemented by Julien Cortial in the open-source finite element framework Albany [40]. This algorithm is similar to that developed for Missing Point Estimation [2].

3.3.3 Projection error

This routine reads in a POD basis and a set of snapshots, projects the snapshots onto the modes, and computes the resulting projection error.

3.3.4 Post processing

This routine reads in a POD basis and a set of generalized coordinates (i.e., modal coefficients) and constructs the resulting flow field. This is useful for recreating the flow when running with a sample mesh.

Chapter 4

Basis stabilization and enhancement via minimal subspace rotation

As described in Section 2.1, a basis for the low-dimensional subspace defining a ROM is obtained from a higher-dimensional subspace through truncation, defined as the removal of modes deemed unimportant in representing the problem solution. Typically, the size of the reduced basis is chosen according to an energy criterion: modes with low energy are discarded, so that the reduced basis consists of the highest energy modes. In most realistic applications, e.g., nonlinear compressible fluid flow, a basis that captures 99% or more of the snapshot energy is required for the ROM to accurately reproduce the snapshots from which it was constructed. Computing a basis that captures 99% of the snapshot energy is in general only feasible for toy problems and/or low resolution models, however, as the “99% energy” requirement leads to relatively high-dimensional bases for problems of interest here. Moreover, higher order modes are often unreliable for prediction, so including them in a reduced basis is unlikely to improve the predictive capabilities of a ROM.

It has been argued by a number of researchers, including Balajewicz and Tezaur, that, for a low-dimensional ROM to be stable and accurate, the truncated/unresolved subspace must be accounted for in some way. Although low energy modes are negligible from a data compression point of view, they are often crucial for representing solutions to the dynamical equations. For fluid flow applications, low energy higher-order modes are associated with energy dissipation. Since low-dimensional ROMs lack the appropriate balance between energy production and energy dissipation, low-dimensional ROMs can be inaccurate and sometimes unstable, as shown for the specific case of POD/Galerkin incompressible and compressible flow ROMs in [9, 22, 5, 4]. While LSPG ROMs tend to possess better stability properties than Galerkin ROMs [14], instabilities have been observed when running low-dimensional ROMs constructed using this method for long-time simulations in the high Reynolds number regime. Sometimes these instabilities result in the calculation of non-physical quantities, e.g., negative energies or pressures, by the ROM.

The situation described above has motivated the development of approaches for stabilizing and enhancing projection-based ROMs by modifying the projection subspace such that it captures more of the low-energy but highly dissipative scales of the flow solution in a way that does not increase the reduced basis size. Building upon earlier work by Balajewicz *et al.* [5] for incompressible flow, we have developed a reduced basis stabilization and enhancement approach for compressible flow that accounts for truncated higher order modes *a priori* via a minimal rotation of the projection subspace [4]. In this method, a low-dimensional reduced basis is stabilized through the offline

solution of a small trace minimization problem on the Stiefel manifold. One important advantage of the method is that it gives rise to a ROM formulation that is consistent with the underlying FOM. Another important advantage is that the method is non-intrusive, meaning it could be applied with ease to a variety of ROMs without the need to modify the ROM source code. The proposed basis rotation approach has been shown to produce projection-based ROMs that are stable and capable of reproducing statistical features of the flow (e.g., pressure PSDs) for long-time simulations in [4].

In this chapter, we briefly review the proposed approach of stabilizing and enhancing low-dimensional ROMs using minimal subspace rotation in the context of a Galerkin ROM applied to the nonlinear compressible flow equations in specific volume form (for completeness). We then describe the method's extension to LSPG ROMs implemented in a flow solver such as SPARC.

4.1 Application of basis rotation to Galerkin ROMs

Consider, without loss of generality¹, the 3D compressible Navier-Stokes equations in primitive, dimensionless, specific-volume form:

$$\begin{aligned} \zeta_{,t} + \zeta_{,j} u_j - \zeta u_{j,j} &= 0, \\ u_{i,t} + u_{i,j} u_j + \zeta p_{,i} - \frac{1}{Re} \zeta \tau_{ij,j} &= 0, \\ p_{,t} + u_j p_{,j} + \gamma u_{j,j} p - \left(\frac{\gamma}{Pr Re} \right) (\kappa(p\zeta)_{,j})_{,j} - \left(\frac{\gamma-1}{Re} \right) u_{i,j} \tau_{ij} &= 0, \end{aligned} \quad (4.1)$$

for $i, j = 1, 2, 3$. Here, the symbol p denotes the fluid pressure, and $\zeta \equiv 1/\rho$ denotes the specific volume of the fluid, where ρ is the fluid density; Pr and Re denote the Prandtl and Reynolds numbers, respectively. The stress tensor τ_{ij} is given by

$$\tau_{ij} = \mu (u_{i,j} + u_{j,i}) + \lambda u_{k,k} \delta_{ij}, \quad (4.2)$$

for $i, j, k = 1, 2, 3$, where δ_{ik} denotes the Kronecker delta. μ and λ are the Lamé coefficients, and the symbol κ denotes the thermal diffusivity, assumed herein to be constant.

In the Galerkin ROM approach, the governing variable, $\mathbf{w}(\mathbf{x}, t)$ is discretized using basis functions (modes) $\{\phi_i(\mathbf{x})\}_{i=1}^M \in H$ with corresponding mode coefficients $\{a_i(t)\}_{i=1}^M$

$$\mathbf{w}(\mathbf{x}, t) \approx \mathbf{w}_0(\mathbf{x}) + \mathbf{w}^{[1..M]}(\mathbf{x}, t) := \mathbf{w}_0(\mathbf{x}) + \sum_{i=1}^M a_i(t) \phi_i(\mathbf{x}), \quad (4.3)$$

where $\mathbf{w}_0(\mathbf{x})$ denotes the (steady) mean flow. A Galerkin projection applied to (4.1) yields a system of coupled quadratic ODEs whose constant coefficients are calculated off-line and once and for all (see Appendix A of [21] for details). This system has the form:

¹We choose to formulate the proposed method in the context of equations expressed in the primitive, rather than conservative, variables so that hyper-reduction can be avoided (all nonlinearities in (4.1) are polynomial). It is possible to extend our proposed approach to problems where the full-order model (FOM) is based on the compressible Navier-Stokes equations in conservative form; see Section 4.2.

$$\frac{d\mathbf{a}}{dt} = \mathbf{C} + \mathbf{L}\mathbf{a} + \begin{bmatrix} \mathbf{a}^T \mathbf{Q}^{(1)} \mathbf{a} & \mathbf{a}^T \mathbf{Q}^{(2)} \mathbf{a} & \dots & \mathbf{a}^T \mathbf{Q}^{(M)} \mathbf{a} \end{bmatrix}^T, \quad (4.4)$$

where $\mathbf{C} \in \mathbb{R}^M$, $\mathbf{L} \in \mathbb{R}^{M \times M}$ and $\mathbf{Q}^{(i)} \in \mathbb{R}^{M \times M}$, $\forall i = 1, \dots, M$. These matrices can be computed using a previously-developed in-house model reduction code known as *Spirit* [8, 22, 17]. *Spirit* is a C++ parallel POD/Galerkin ROM code that reads in snapshot data written by a high-fidelity code, computes a POD basis from this snapshot set using the *Anasazi* package from the *Trilinos* suite of libraries [19], and constructs a ROM by projecting the governing linear or nonlinear fluid PDEs onto the reduced basis in one of several inner products. For a detailed discussion of the *Spirit* code, the reader is referred to [22, 17].

4.1.1 Trace minimization problem for subspace rotation

The key idea behind the minimal subspace rotation approach is to model the truncated modes *a priori* by “rotating” the projection subspace into a more dissipative regime. Specifically, instead of approximating the solution using only the first M most energetic POD modes, the solution is approximated using a linear-superposition of $M + P$ (with $P > 0$) most energetic POD modes. Mathematically this can be expressed as:

$$\tilde{\phi}_i = \sum_{j=1}^{M+P} X_{ji} \phi_j \quad i = 1, \dots, M, \quad (4.5)$$

where $\mathbf{X} \in \mathbb{R}^{(M+P) \times M}$ is the orthonormal ($\mathbf{X}^T \mathbf{X} = \mathbf{I}_{M \times M}$) “rotation” matrix. The Galerkin system tensors associated with these new modes are expressed as a function of \mathbf{X} as follows:

$$\tilde{Q}_{jk}^{(i)} = \sum_{s,q,r=1}^{M+P} X_{si} Q_{qr}^{(s)} X_{qj} X_{rk} \quad i, j, k = 1, \dots, M, \quad (4.6a)$$

$$\tilde{\mathbf{L}} = \mathbf{X}^T \mathbf{L} \mathbf{X}, \quad (4.6b)$$

$$\tilde{\mathbf{C}} = \mathbf{X}^T \mathbf{C}^*, \quad (4.6c)$$

where $\mathbf{C} \in \mathbb{R}^{M+P}$, $\mathbf{L} \in \mathbb{R}^{(M+P) \times (M+P)}$ and $\mathbf{Q}^{(i)} \in \mathbb{R}^{(M+P) \times (M+P)}$, $\forall i = 1, \dots, (M+P)$ are the Galerkin system coefficients corresponding to the first $M + P$ most energetic POD modes. The new Galerkin system is hence:

$$\frac{d\mathbf{a}}{dt} = \tilde{\mathbf{C}} + \tilde{\mathbf{L}}\mathbf{a} + \begin{bmatrix} \mathbf{a}^T \tilde{\mathbf{Q}}^{(1)} \mathbf{a} & \mathbf{a}^T \tilde{\mathbf{Q}}^{(2)} \mathbf{a} & \dots & \mathbf{a}^T \tilde{\mathbf{Q}}^{(M)} \mathbf{a} \end{bmatrix}^T, \quad (4.7)$$

where the matrices $\tilde{\mathbf{Q}}^{(i)}$, $\tilde{\mathbf{L}}$ and $\tilde{\mathbf{C}}$ are given by (4.6). Remark that (4.6) is consistent with the original ROM formulation (4.4), and hence the governing PDEs (4.1). This is not the case for a

popular class of methods for stabilizing ROMs, termed eddy-viscosity-based stabilization methods [45, 35]. In these approaches, the dynamics of the truncated modes are modeled by modifying the constant and linear terms of the Galerkin model to incorporate an eddy-viscosity based closure model into the ROM².

The goal of the proposed approach is to find \mathbf{X} such that:

- (i) the new modes $\tilde{\boldsymbol{\phi}}_i$ remain good approximations of the flow, and
- (ii) the new Galerkin ROM is stable and accurate.

Once the goals (i) and (ii) are translated into mathematical statements, a constrained optimization problem is formulated for \mathbf{X} . To guarantee that the new modes remain good approximations of the flow, the distance $\|\mathbf{X} - \mathbf{I}_{M+P,M}\|_F$ is minimized, where $\mathbf{I}_{M+P,M}$ are the first M columns of an $M+P$ identity matrix, and $\|\cdot\|_F$ is the Frobenious norm. To ensure that the ROM is stable and accurate, an eddy-viscosity-based constraint equation is used as a proxy. In other words, the eigenvalues of the linear operator are modified until a stable and accurate ROM is generated.

Mathematically, the constrained optimization problem for \mathbf{X} outlined above reads as follows:

$$\begin{aligned} & \underset{\mathbf{X} \in \mathcal{V}_{(M+P),M}}{\text{minimize}} && -\text{tr}(\mathbf{X}^T \mathbf{I}_{(M+P) \times M}) \\ & \text{subject to} && \text{tr}(\mathbf{X}^T \mathbf{L} \mathbf{X}) = \eta \end{aligned} \quad (4.8)$$

where $\eta \in \mathbb{R}$ and

$$\mathcal{V}_{(M+P),M} \in \{\mathbf{X} \in \mathbb{R}^{(M+P) \times M} : \mathbf{X}^T \mathbf{X} = \mathbf{I}_M, P > 0\}. \quad (4.9)$$

In (4.9), $\mathcal{V}_{(M+P),M}$ is the Stiefel manifold, defined as the set of $(M+P) \times M$ matrices satisfying the orthonormality condition $\mathbf{X}^T \mathbf{X} = \mathbf{I}_M$ [33, 43]. In (5.7), the objective function is simplified by utilizing the property that for a real matrix $\|\mathbf{A}\|_F^2 = \text{tr}(\mathbf{A}^T \mathbf{A})$. Thus, minimizing $\|\mathbf{X} - \mathbf{I}_{M+P,M}\|_F$ is equivalent to minimizing $-\text{tr}(\mathbf{X}^T \mathbf{I}_{(M+P) \times M})$.

The appropriate eigenvalue distribution η , must be identified using a solution matching procedure. Our approach for selecting η is discussed in Section 4.1.3

4.1.1.1. Alternative objective functions and constraints

There are other choices for the objective function and constraint in (5.7) than the ones we have selected. The optimization problem (5.7) can be formulated more generally as follows:

$$\begin{aligned} & \underset{\mathbf{X} \in \mathcal{V}_{(M+P),M}}{\text{minimize}} && f(\mathbf{X}) \\ & \text{subject to} && g(\mathbf{X}, \mathbf{L}) = 0, \end{aligned} \quad (4.10)$$

for some goal-oriented objective function $f(\mathbf{X})$ and constraints $g(\mathbf{X}, \mathbf{L}) = 0$. Possible objective functions in (4.10) include:

²For a review of eddy-viscosity-based stabilization methods for model reduction, the reader is referred to [4].

- Maximize resolved turbulent kinetic energy (TKE)³:

$$f(\mathbf{X}) = -\|\boldsymbol{\Sigma} - \mathbf{X}\mathbf{X}^T\boldsymbol{\Sigma}\|_F, \quad (4.11)$$

where $\boldsymbol{\Sigma}$ denotes the square of the second moments of the ROM modal coefficients [5].

- Minimize weighted subspace rotation:

$$f(\mathbf{X}) = \|\mathbf{W}(\mathbf{X} - \mathbf{I}_{M+P,M})\|_F, \quad (4.12)$$

where \mathbf{W} is some matrix of weights.

- Maximize parametric robustness:

$$f(\mathbf{X}) = -\sum_{i=1}^Q \beta_i \|\Phi_M^*(\mu_i)\mathbf{X} - \Phi_M^*(\mu_i)\|_F, \quad (4.13)$$

where β_i are weights and μ_i are parameters (e.g., Reynolds or Mach number), for $i = 1, \dots, Q$. Here, Φ_M^* denotes the original (unrotated) basis.

- Minimize Ordinary differential equation (ODE) based objective:

$$f(\mathbf{X}, \mathbf{L}) = \|\mathbf{a}(t) - \tilde{\mathbf{a}}(t)\| \quad (4.14)$$

where $\tilde{\mathbf{a}}(t)$ is a reference time series for ROM modal amplitudes $\mathbf{a}(t)$.

Some candidates for the constraint in (4.10) include:

- Manipulation of individual eigenvalues:

$$g(\mathbf{X}, \mathbf{L}) = \tilde{\lambda}_i - \beta_i, \quad (4.15)$$

for some $\beta_i \in \mathbb{R}$, where $\tilde{\lambda}_i$ denote the eigenvalues of $\tilde{\mathbf{L}}$.

- Linear energy conservation:

$$g(\mathbf{X}, \mathbf{L}) = \tilde{\mathbf{L}}_H, \quad (4.16)$$

where $\tilde{\mathbf{L}}_H$ is the Hermitian part of $\tilde{\mathbf{L}}$.

4.1.2 Solution of constrained optimization problem

A common method for solving constrained optimization problems of the form (5.7) is the method of Lagrange multipliers [32]. In this method, the Lagrangian of the optimization problem is computed, and its stationary points are sought, yielding necessary optimality conditions for local maxima and minima. The reader can verify that the Lagrangian for (5.7) is

$$\mathcal{L}(\mathbf{X}, \boldsymbol{\Lambda}_1, \boldsymbol{\Lambda}_2) := -\text{tr}(\mathbf{X}^T \mathbf{I}_{(M+P) \times M}) + \text{tr}(\boldsymbol{\Lambda}_1 (\mathbf{X}^T \mathbf{L}^* \mathbf{X} - \frac{\eta}{M} \mathbf{I}_M)) + \text{tr}(\boldsymbol{\Lambda}_2 (\mathbf{X}^T \mathbf{X} - \mathbf{I}_m)), \quad (4.17)$$

³Maximizing TKE was considered in earlier work involving basis stabilization via subspace rotation for incompressible flow [5]. For both incompressible as well as compressible flow, maximizing this objective functional led to less accurate results than minimizing the subspace rotation.

where Λ_1 and Λ_2 are diagonal matrices of Lagrange multipliers.

Suppose that \mathbf{X} is a local maximizer of problem (5.7). Then \mathbf{X} satisfies the first-order optimality condition $\mathcal{L}_{\mathbf{X}} = -\mathbf{I}_{(M+P) \times M} + \Lambda_1(\mathbf{L}^* + \mathbf{L}^{*\top})\mathbf{X} + 2\Lambda_2\mathbf{X} = \mathbf{0}$, $\text{tr}(\mathbf{X}^T\mathbf{L}^*\mathbf{X} - \frac{\eta}{M}\mathbf{I}_M) = \mathbf{0}$, and $\mathbf{X}^T\mathbf{X} - \mathbf{I}_M = \mathbf{0}$. Solving (5.7) using Lagrange multipliers is possible; however, it is inefficient. Significant speed-ups are possible by satisfying the orthonormality constraint directly via optimization on the Stiefel matrix manifold. In this method, with the help of the augmented Lagrange method, the constrained optimization problem is reduced to an unconstrained optimization problem on the Stiefel manifold as follows:

$$\underset{\mathbf{X} \in \mathcal{Y}_{(M+P),M}}{\text{minimize}} \quad -\text{tr}(\mathbf{X}^T\mathbf{I}_{(M+P) \times M}) + \frac{\mu_k}{2}\text{tr}(\mathbf{X}^T\mathbf{L}^*\mathbf{X} - \frac{\eta}{n}\mathbf{I}_M)^2 - \lambda_{\mathcal{L}}\text{tr}(\mathbf{X}^T\mathbf{L}^*\mathbf{X} - \frac{\eta}{n}\mathbf{I}_M), \quad (4.18)$$

where μ_k is increased until the constraint is satisfied to some desired precision. The variable $\lambda_{\mathcal{L}}$ is an estimate of the Lagrange multiplier and is updated according to the rule

$$\lambda_{\mathcal{L}i+1} \leftarrow \lambda_{\mathcal{L}i} - \mu_k \text{tr}(\mathbf{X}^{(k)T}\mathbf{L}^*\mathbf{X}^{(k)} - \frac{\eta}{M}\mathbf{I}_M), \quad (4.19)$$

where $\mathbf{X}^{(k)}$ is the solution of the unconstrained problem at the k^{th} step. In this work, the Manopt MATLAB toolbox [10] is used to solve (4.18).

4.1.3 Stabilization algorithm

The overall stabilization procedure is summarized in Algorithm 1. The distribution of stabilizing eigenvalues η in (4.18) is identified using a brute force approach, which is feasible in this case because the associated ROMs are small and relatively inexpensive to integrate numerically. A global ROM “energy” is defined as follows: $E(t)^{(k)} = \sum_i^M (a(t)_i^{(k)})^2$. A linear fit of the temporal data is performed: $E(t)^{(k)} \approx c_1^{(k)}t + c_0^{(k)}$. If the absolute value of $c_1^{(k)}$ is below some predefined tolerance, the ROM is deemed stable and the procedure halts. If the value of $c_1^{(k)}$ is positive (indicating increasing energy) then $\eta^{(k+1)} = \eta^{(k)} + \varepsilon$, where $\varepsilon < 0$. If the value of $c_1^{(k)}$ is negative (indicating decreasing energy) then $\eta^{(k+1)} = \eta^{(k)} + \varepsilon$ where $\varepsilon > 0$. This search is automated using MATLAB’s `fzero` root finding algorithm.

For a more detailed discussion of the stabilization algorithm, including some guidelines for selecting the parameters in Algorithm 1, and some general remarks about the numerical solution of the optimization problem (5.7), the reader is referred to [4].

Algorithm 1: Stabilization algorithm for compressible Navier-Stokes equations

input : Initial guess for stabilizing eigenvalue distribution $\eta^{(0)}$, ROM size M and $P \geq 1$, Galerkin system matrices associated with the first $M + P$ most energetic POD modes. $\mathbf{C} \in \mathbb{R}^{M+P}$, $\mathbf{L} \in \mathbb{R}^{(M+P) \times (M+P)}$ and $\mathbf{Q}^{(i)} \in \mathbb{R}^{(M+P) \times (M+P)}$, $i = 1, \dots, M + P$, convergence tolerance TOL , and maximum number of iterations. k_{max}

output: Stabilizing rotation matrix \mathbf{X}

1 **for** $k = 0, \dots, k_{max}$ **do**

2 Solve constrained optimization problem on Stiefel manifold:

$$\begin{aligned} & \underset{\mathbf{X}^{(k)} \in \mathcal{V}_{(M+P), M}}{\text{minimize}} && -\text{tr} \left(\mathbf{X}^{(k)\text{T}} \mathbf{I}_{(M+P) \times M} \right) \\ & \text{subject to} && \text{tr}(\mathbf{X}^{(k)\text{T}} \mathbf{L} \mathbf{X}^{(k)}) = \eta^{(k)}. \end{aligned}$$

3 Construct new Galerkin matrices using (4.6)

4 Integrate numerically new Galerkin system

5 Calculate modal energy $E(t)^{(k)} = \sum_i^M (a(t)_i^{(k)})^2$

6 Perform linear fit of temporal data $E(t)^{(k)} \approx c_1^{(k)} t + c_0^{(k)}$

7 Based on energy growth $c_1^{(k)}$, calculate ε using root-finding algorithm and perform update $\eta^{(k+1)} = \eta^{(k)} + \varepsilon$

8 **if** $\|c_1^{(k)}\| < TOL$ **then**

9 $\mathbf{X} := \mathbf{X}^{(k)}$

10 **terminate the algorithm**

11 **end**

12 **end**

4.2 Application of basis rotation to LSPG ROMs

Having formulated the minimal subspace rotation approach for reduced basis stabilization and enhancement in the context of Galerkin ROMs in Section 4.1, we now discuss the method's extension to LSPG ROMs (Section 2.2.2). For concreteness and without loss of generality, we focus on the specific case in which the LSPG ROM is computed using the `SPARC` code. Our initial stabilization workflow depends also on the `Spirit` code⁴ (for generating the matrices and tensors appearing in (4.4)) and the `Manopt` MATLAB toolbox (for solving the optimization problem (4.18)). The workflow is as follows:

- (i) Run `SPARC` in FOM mode to collect snapshots.
- (ii) Run `SPARC` in ROM mode to generate a POD basis.
- (iii) Transform the POD basis from `SPARC` to `Spirit`. In particular⁵:
 - Interpolate the POD modes from element centers (`SPARC` convention) to nodes (`Spirit` convention).
 - Transform the POD modes from conservative variables (used in `SPARC`) to primitive variables (used in `Spirit`).
- (iv) Run `Spirit` with the POD basis from step (iii) to generate the matrices \mathbf{L} , $\mathbf{Q}^{(i)}$ and \mathbf{C} appearing in (4.4), as well as a mass matrix \mathbf{M} pre-multiplying the transient term in (4.4)⁶.
- (v) Apply Algorithm 1 from Section 4.1.3 to stabilize the `SPARC` POD modes using the matrices \mathbf{L} , $\mathbf{Q}^{(i)}$, \mathbf{C} and \mathbf{M} from `Spirit`.
- (vi) Transform the stabilized basis back to `SPARC` variables (reverse of step (iii)).
- (vii) Run `SPARC` in ROM mode with the scaled stabilized POD basis instead of the original POD basis.

The workflow described above should be invoked in the case a `SPARC` ROM for a particular problem of interest experiences instabilities. This is not the only situation in which the approach can be beneficial, however. As explained at the beginning of this chapter, the proposed basis stabilization and enhancement methodology can enable extreme model reduction by producing accurate ROMs of very small dimension. The approach is therefore recommended if one is interested in reducing the dimension of a ROM without sacrificing accuracy or stability.

Once the basic workflow described above has been put in place, verified and demonstrated, one may consider modifying the objective function and constraints in (5.7), for example, as suggested in Section 4.1.1.1.

⁴See Section 4.1 and [8, 22, 17].

⁵Some additional scaling and/or normalization of the modes may be required in `Spirit` to avoid bad scaling of the ROM matrices and tensors.

⁶A mass matrix arises because the POD modes from `SPARC` scaled following step (ii) will not be orthonormal.

Chapter 5

Numerical results

This chapter summarizes the results for two numerical examples aimed at determining if accurate predictive ROMs can be constructed for captive-carry problems using the MOR suite implemented within the SPARC code (see Chapter 3 for details of the implementation). Two test cases are considered: a 2D viscous laminar cavity problem, and the problem of an inviscid pulse in a uniform base flow. In Section 5.1, we evaluate the performance of the LSPG ROMs in SPARC on these test cases. Section 5.2 aims to gauge the viability of the basis stabilization approach detailed in Chapter 4 on problems akin to those considered in Section 5.1.

Before presenting the main results, two remarks are in order.

First, the examples considered here are admittedly much simpler than the transonic, turbulent flow simulations we are ultimately interested in. Once we have confidence that accurate ROMs can be constructed for the simplified tests considered here, we will evaluate our ROM methodology on higher Reynolds number, turbulent problems.

Second, we note that the implementation of hyper-reduction in SPARC has been put on hold, as it was deemed more important to ensure that the ROMs in SPARC are viable without hyper-reduction, which can introduce additional error (Section 3.2). For this reason, our SPARC ROMs do not yet achieve the speedups possible with hyper-reduction.

5.1 Numerical evaluation of LSPG ROMs in SPARC

5.1.1 Viscous laminar cavity

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 5.1. A formal mesh convergence study is planned but has not been completed yet. 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

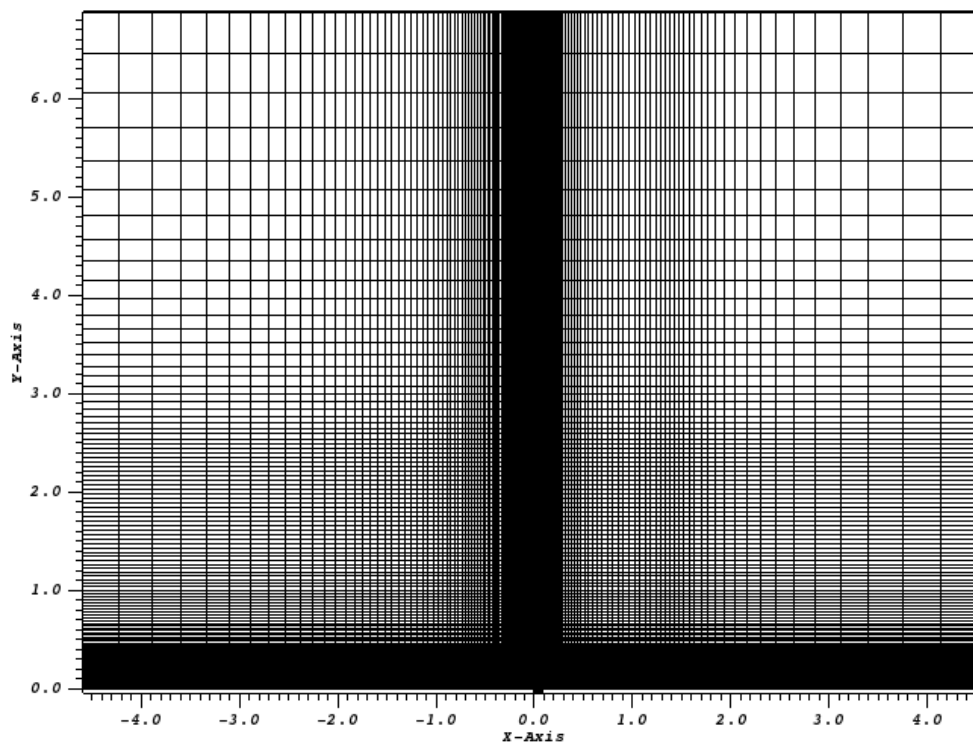


Figure 5.1. The computational mesh used for 2D viscous laminar cavity simulations in SPARC.

Parameter	Dimensional Value
Free-stream Velocity, u_1	$208.7816 m/s$
Temperature, T	$300 K$
Density, ρ	$2.90 \times 10^{-4} kg/m^3$
Viscosity, μ	$8.46 \times 10^{-7} kg/(ms)$
Specific Gas Constant, R	$287.097384766765 m^2/(s^2 K)$
γ	1.4
Prandtl Number	0.72

Table 5.1. Parameters used for the 2D viscous laminar cavity test case in SPARC.

boundaries were seen to have a significant impact on the accuracy and stability of ROMs in previous work [22]. 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 5000. The exact parameters specified in the SPARC input file are given in Table 5.1. The nominal full-order model runs used BDF2 time stepping with a fixed time step of $2 \times 10^{-7} s$ which corresponds to a CFL number of under 5.0.

We are interested in the pressure fluctuations inside the cavity. The pressure is output every time step at a location midway up the downstream wall of the cavity. The resulting time history can then be used to compute the PSD of the pressure fluctuations.

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 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.

Three types of ROM runs are discussed in this section. First, there are reproductive ROMs in which the ROM attempts to reproduce the snapshots from the FOM used to create the POD basis. Second, there are runs that use the same basis as the reproductive ROMs but attempt to integrate in time beyond the snapshot collection period. Third, there are predictive runs where snapshots from FOM runs with two different Mach numbers are used to create a basis for a ROM run at an intermediate Mach number.

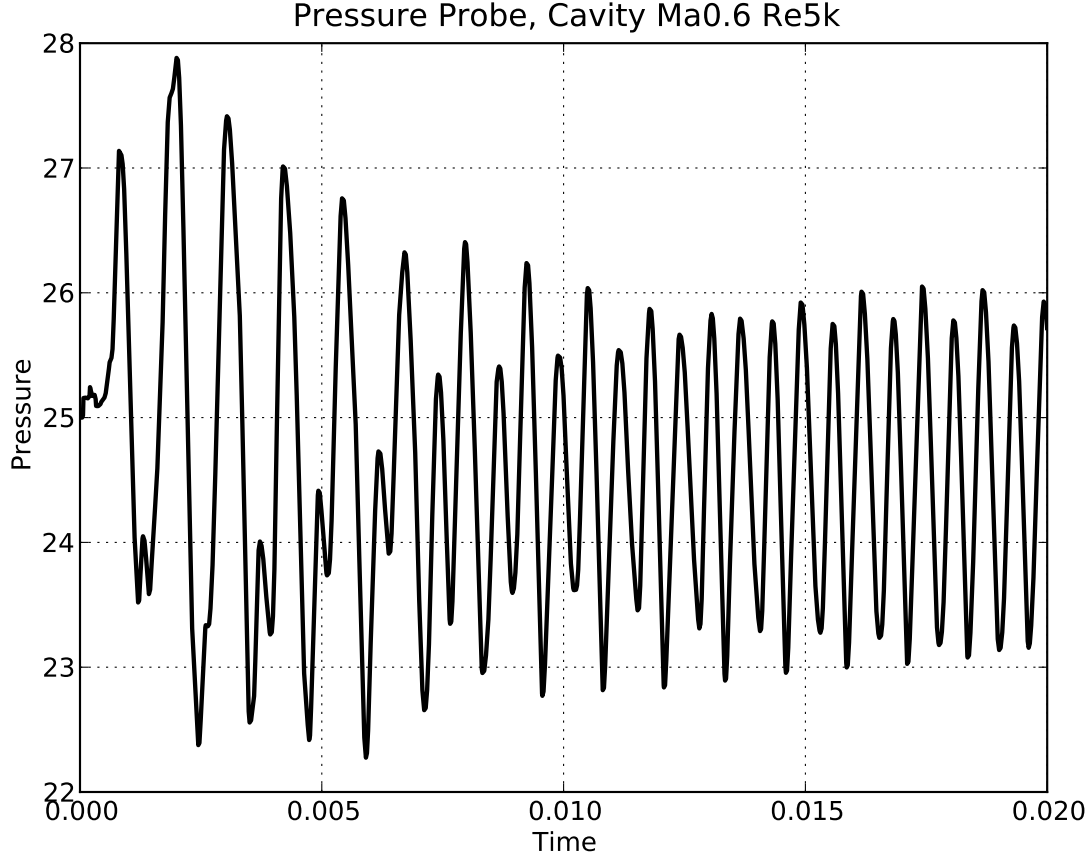


Figure 5.2. The pressure time history for a point midway up the downstream wall of the cavity.

Reproductive ROMs

For these simulations, SPARC is first run using the regular full-order model implementation for 100,000 time steps using a time step of $2 \times 10^{-7} s$. Figure 5.2 shows the pressure time history at a point midway up the downstream cavity wall. An almost periodic region can be identified from this plot beginning a little before $t = 0.15 s$. The FOM was then restarted from $t = 0.15 s$ in order to collect snapshots of the flow field every time step. In order to speed up the resulting ROM calculations, the time step for this FOM run was changed to $2 \times 10^{-6} s$. Changing the time step did not seem to impact the FOM runs very much, but it reduced the number of snapshots by a factor of 10.

Next, a POD basis was constructed using all of the snapshots that were collected. A single basis was constructed for all the conservative flow variables: density, x -momentum, y -momentum, z -momentum, and energy. The initial snapshot, i.e., at $t = 0.15 s$, was used as the base flow and was subtracted from all of the other snapshots prior to computing the basis. No other pre-processing, such as non-dimensionalization or normalization, was performed.

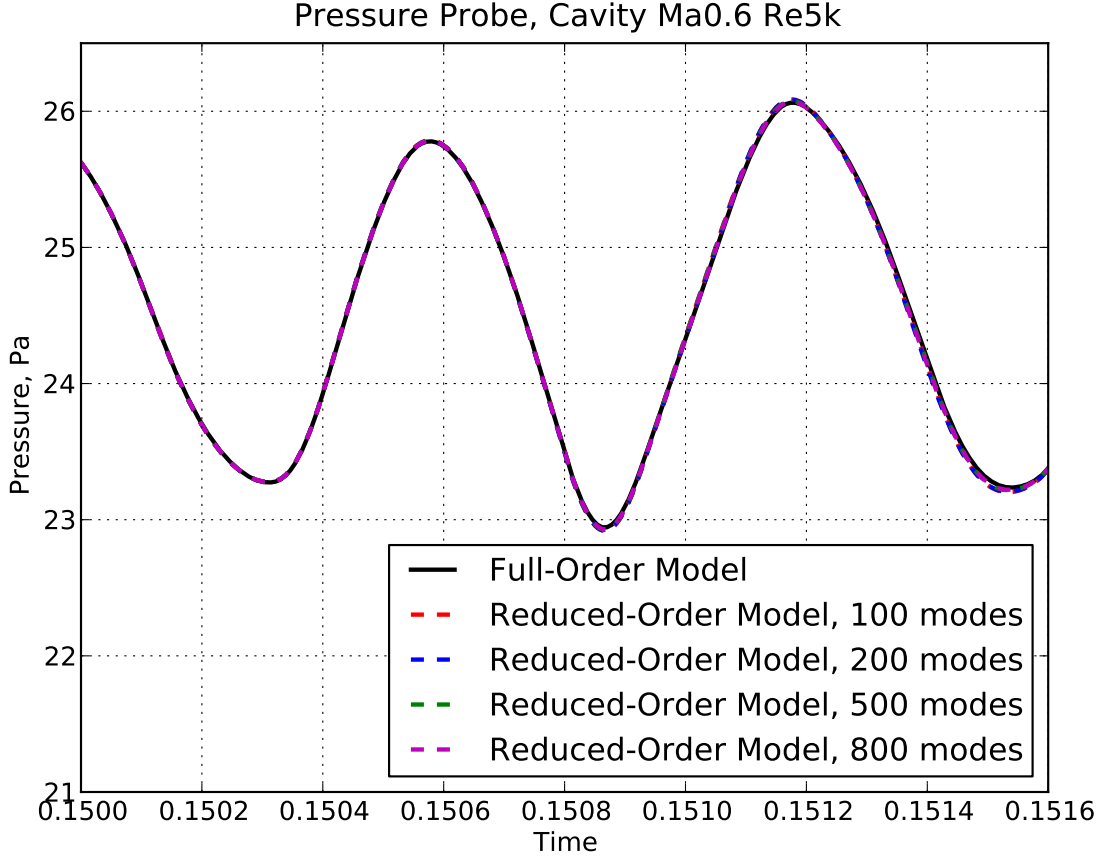


Figure 5.3. The pressure time histories at a point midway up the downstream wall of the cavity for the FOM and reproductive ROMs using a POD basis constructed from 800 snapshots.

Short-time simulation

First, we ran SPARC in ROM mode with implementation with a POD basis constructed using 800 snapshots (approximately one full period of variation). Figure 5.3 shows a comparison of the pressure time histories at a point midway up the downstream wall of the cavity. This plot compares the FOM and the reproductive ROMs using varying numbers of modes. Although they are not perfect, the ROMs appear to reproduce the FOM pressure history fairly accurately.

Figure 5.4 shows the discarded energy fractions for the POD basis constructed from 800 snapshots. The discarded energy fraction is the amount of energy that will be neglected when truncating the basis to a finite number of modes. Using 200 modes, the POD basis captures over 99.99% of the energy contained in the snapshot set. Using 500 modes captures over 99.999% of the energy, and using all 800 modes captures 100% of the energy.

Figure 5.5 shows a plot of the projection error for the basis constructed from 800 snapshots. This is computed by taking the snapshots from a full-order model run, projecting them onto a basis of

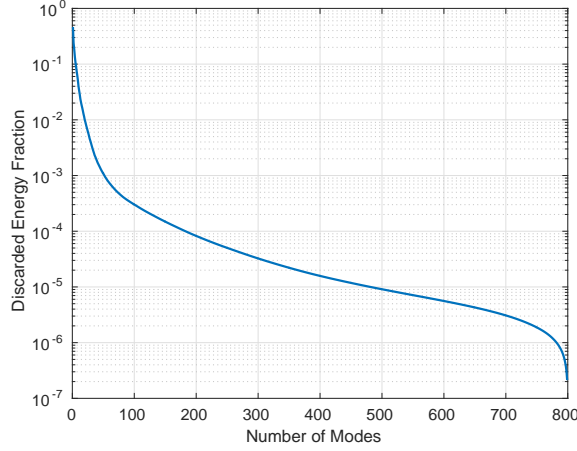


Figure 5.4. The discarded energy fractions for the POD basis constructed from 800 snapshots for the cavity problem.

a given size, then reconstructing the flow field and computing the error between it and the FOM. The dashed line on the plot indicates the end of the snapshot collection period. Prior to this time, the projection errors are fairly low and the flow field can be seen that the error decreases as more modes are used. When all 800 modes are used the projection error is under 1.0×10^{-14} and close to machine precision up until the end of the snapshot collection period. Beyond the snapshot collection period, the projection error jumps to a higher value for all but the smallest numbers of modes considered. This indicates that there may be accuracy issues when using this basis for predictive ROM runs beyond the snapshot collection period, as will be discussed in a later section. Prior to the end of the snapshot collection period, the fact that the projection error when using 800 modes is close to machine precision indicates that an 800 mode basis is capable of reproducing the FOM snapshots to within machine precision.

Longer-time simulation

A second reproductive test was performed in which the FOM was restarted from $t = 0.15s$ and run for 2000 time steps (approximately 2.5 periods of variation). Figure 5.6 shows the pressure time histories for these runs. For these runs, the ROMs produce pressure variations that grow in magnitude. This may indicate a possible instability in the ROMs. The reader can observe that, even when using all 2000 modes, the ROM fails to reproduce the FOM exactly. This is inconsistent with LSPG ROM theory, which says that a ROM with a full basis ($M = K$) should have zero error. It is conjectured that our ROM has non-zero error when $M = K = 2000$ due to poor scaling, and/or an improper treatment of boundary condition terms in the ROM. The former theory is studied in Section 5.1.2 in the context of a simpler problem involving an inviscid pulse in a uniform base flow. Since SPARC is a dimensional code and the values of the parameters considered are different by up to 5 orders of magnitude (e.g., $\mathcal{O}(10^{-4})$ for density and $\mathcal{O}(10^1)$ for pressure; see Table 5.1), catastrophic cancellation during the projection step of the model reduction and/or inaccurate solver performance due to poor matrix conditioning is conceivable.

Figure 5.7 shows the discarded energy fractions for the POD basis constructed from 2000 snap-

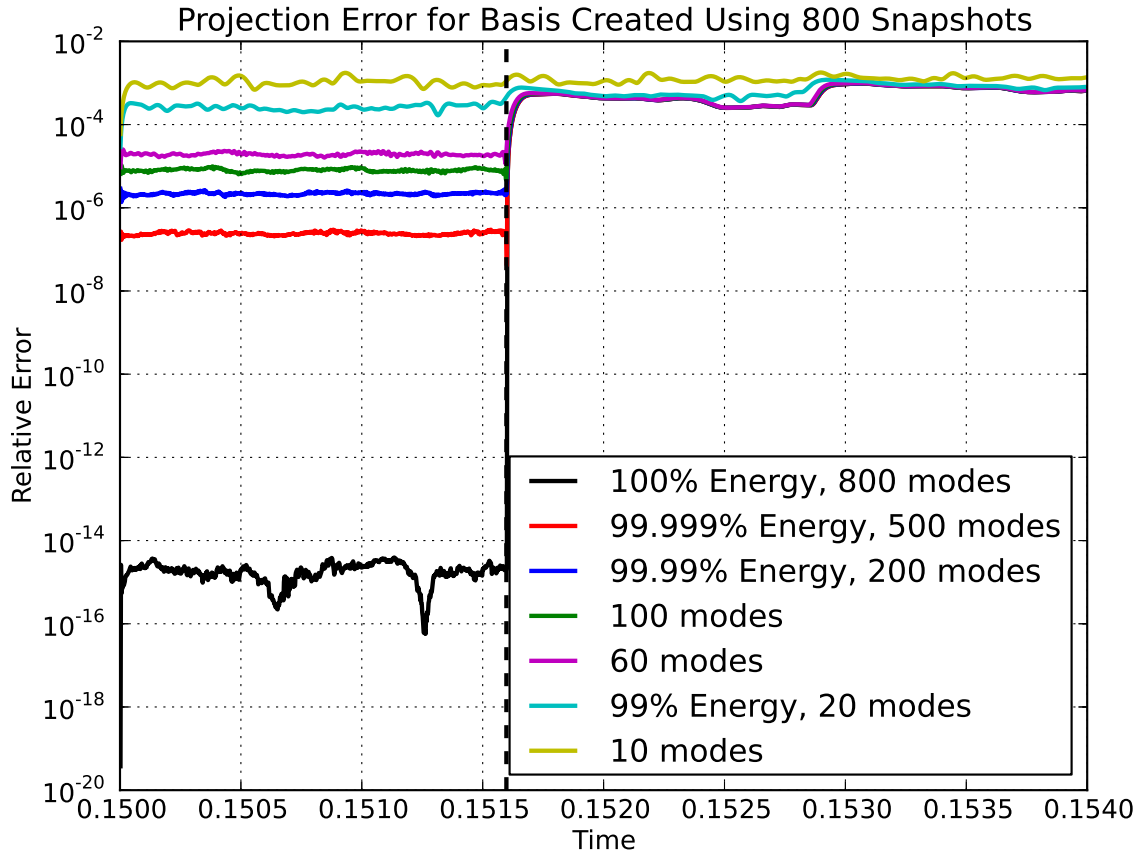


Figure 5.5. The projection error for the POD basis constructed from 800 snapshots for the cavity problem. The dashed line indicates the end of the snapshot collection period.

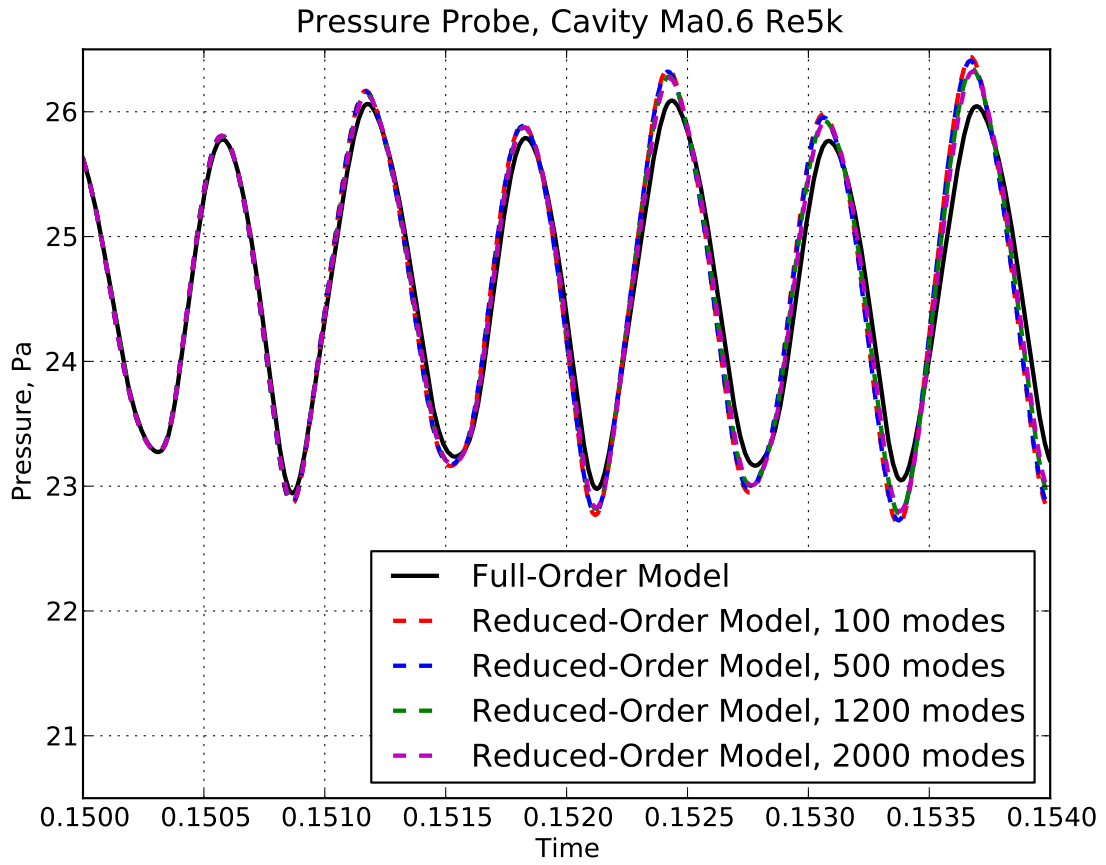


Figure 5.6. The pressure time histories at a point midway up the downstream wall of the cavity for the FOM and reproductive ROMs using a POD basis constructed from 2000 snapshots.

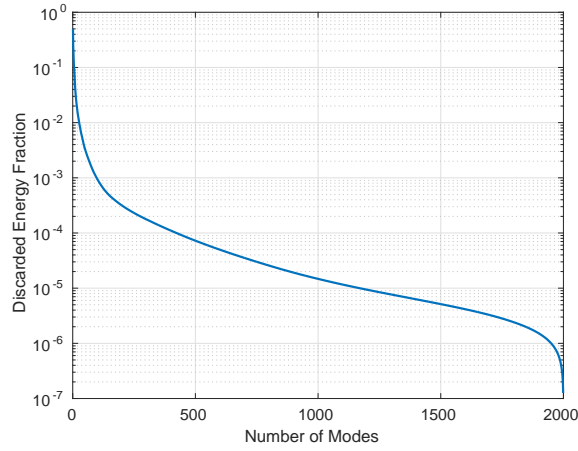


Figure 5.7. The discarded energy fractions for the POD basis constructed from 2000 snapshots for the cavity problem.

shots. Using 100 modes, the POD basis captures over 99.9% of the energy contained in the snapshot set. Using 500 modes captures over 99.99% of the energy, 1200 modes captures over 99.999% of the energy, and using all 2000 modes captures 100% of the energy.

Figure 5.8 shows a plot of the projection error for the basis constructed from 2000 snapshots. The projection errors behave as expected, with the error decreasing as the number of modes used is increased. When using 100 modes, the projection error is under 1.0×10^{-4} . This decreases to under 1.0×10^{-5} for 500 modes, and further to under 1.0×10^{-6} for 1200 modes. When all 2000 modes are used the projection error is under 1.0×10^{-14} and close to machine precision. The fact that the projection error when using 2000 modes is close to machine precision indicates that a 2000 mode basis is capable of reproducing the FOM snapshots to within machine precision.

Figure 5.9 shows a comparison of the power spectral densities for the pressure time histories given in Figure 5.6. The calculation of the PSD for such a short time signal may not be realistic, but the plots are provided here to reinforce that the goal of the ROMs is to reproduce the statistical nature of the cavity flow, not the instantaneous flow features. As can be seen, the ROMs accurately reproduce the FOM PSD for this case. This is to be expected since the observed differences between the ROMs and the FOM had to do with the magnitude of the pressure variations, not the frequencies. The vertical dashed lines indicate the primary frequencies of the FOM pressure signal computed manually, and the PSDs seem to capture these fairly well given the short duration of the time signal.

One possible explanation for the discrepancies between the ROM runs and the FOM may have to do with the tolerances used in SPARC for this case. Runs with tighter tolerances seemed to eliminate the increasing amplitudes, but introduced an apparent shift in frequency, as shown in Figure 5.10. However, this still uses the same basis as the original run, which was based off of a FOM run with the original nominal tolerances. FOM runs with tighter tolerances, and an updated version of SPARC produce time histories such as that shown in Figure 5.11. For this run, the pressure time history does not reach a periodic state. It is possible that it may reach one if run for a longer time, but if not then some other strategy for determining when to start collecting snapshots will be

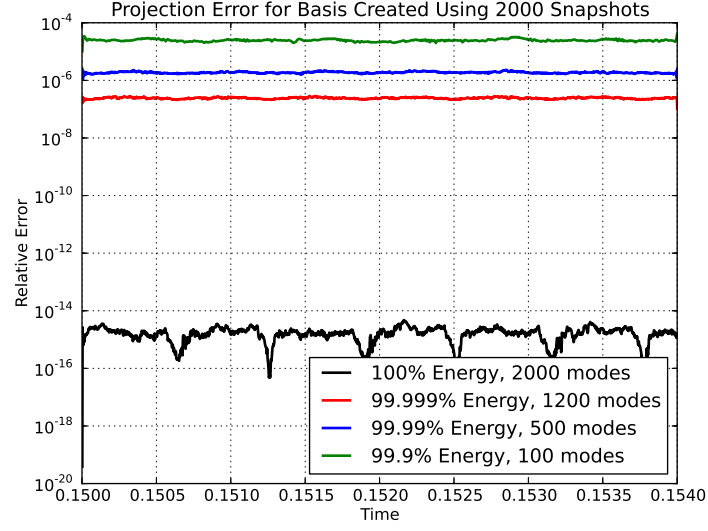


Figure 5.8. The projection error for the POD basis constructed from 2000 snapshots for the cavity problem.

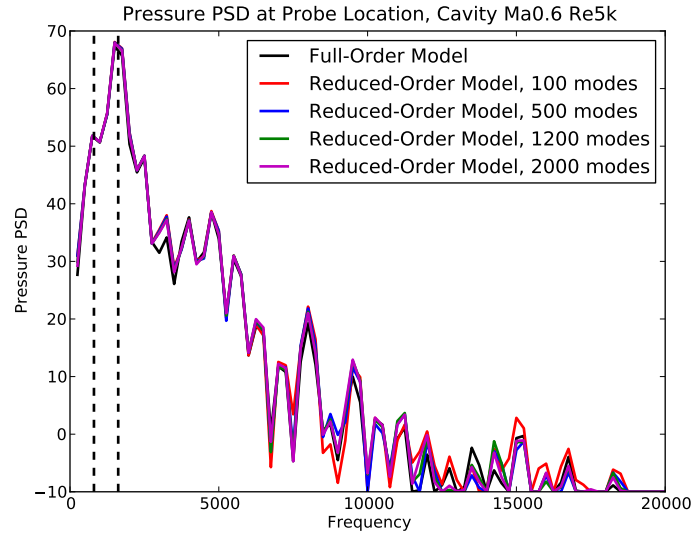


Figure 5.9. The power spectral density of the pressure fluctuations measured at a point midway up the downstream wall of the cavity.

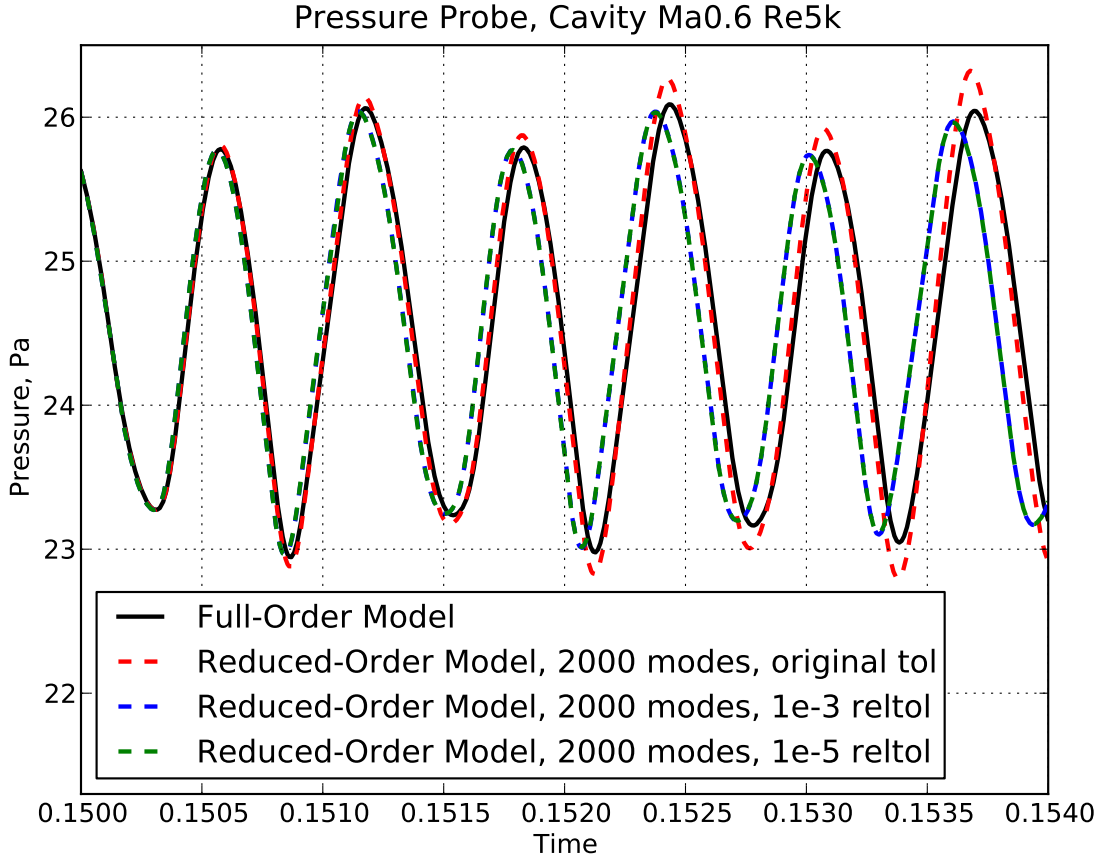


Figure 5.10. The pressure time histories at a point midway up the downstream wall of the cavity for the FOM and reproductive ROMs using a POD basis constructed from 2000 snapshots when the tolerances are varied.

required.

Predictive ROM - Extended Time

Accurately reproducing the FOM is a necessary step in verifying the ROM implementation in SPARC. However, it is also desired that the ROMs can be used to predict behavior for which they were not trained. One desired predictive capability is the ability to extend the time of the simulation beyond the snapshot collection period of the training data.

For this case, the basis created for the reproductive ROMs using 800 snapshots was used to attempt to run for 2000 time steps. Figure 5.12 shows the pressure histories for this case. The vertical dashed line in the plot indicates the end of the snapshot collection period. As discussed earlier, although they are not exact, the ROMs reproduce the FOM fairly accurately up until the

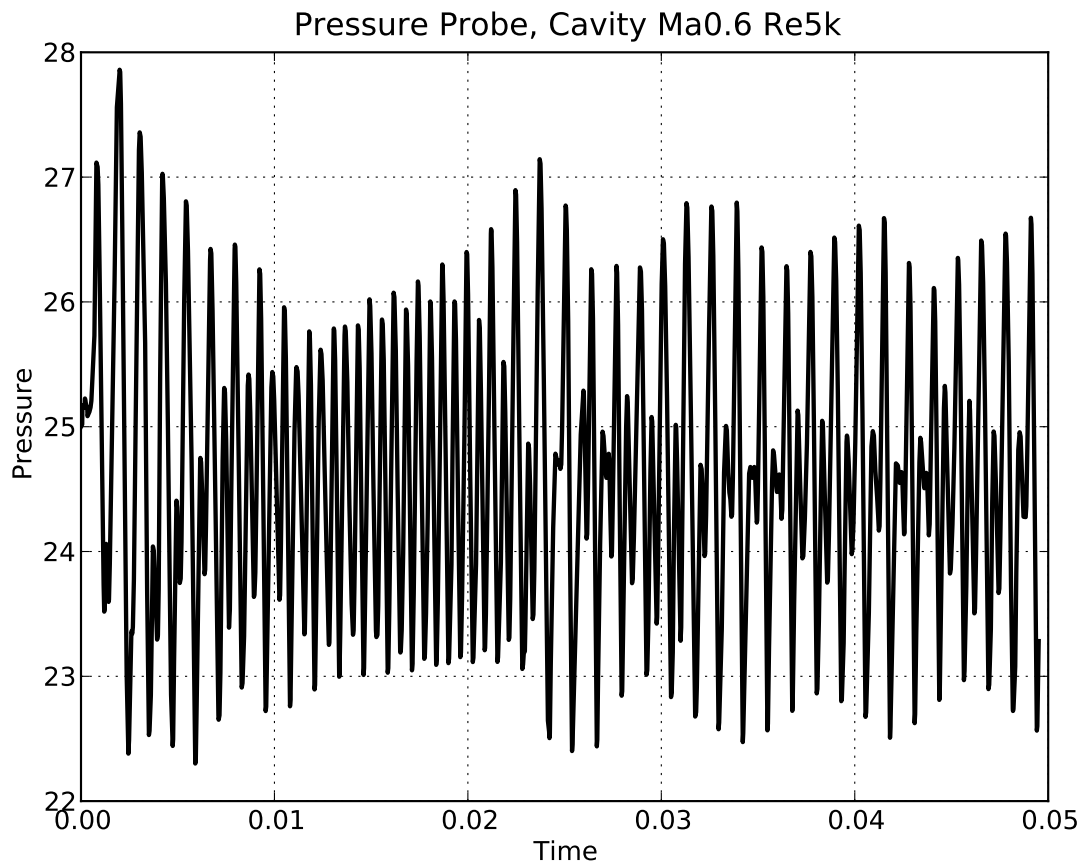


Figure 5.11. The pressure time history for a point midway up the downstream wall of the cavity when an updated version of SPARC is run with tighter tolerances and for a longer period of time. The pressure does not appear to reach a periodic state.

snapshot collection period. However, after this time, the error in the ROMs grows, which may be expected given the projection errors shown in Figure 5.5. The simulations eventually start producing non-physical results, such as negative energies or pressures. This may indicate an instability, or there could be other issues with the ROMs. Before a definitive conclusion about the predictive capabilities of the LSPG ROMs in SPARC can be made, however, the unexpected behavior involving reproductive LSPG ROMs in SPARC (e.g., non-zero error when $M = K$) must be resolved.

Predictive FOM - Mach Number Variation

As discussed above, one of the ultimate goals of the project summarized in this report is to create ROMs that are predictive for use in uncertainty quantification. It is therefore a desired property that the ROMs maintain their accuracy when run at conditions for which they were not trained.

As a simple test of this capability, SPARC was run using the regular full-order model implementation at Mach 0.575 and Mach 0.625. The ROM will then be run at Mach 0.6 and compared to the FOM. For these runs, SPARC ran for 100,000 time steps using a time step of 2×10^{-7} s and snapshots were collected every 50 time steps for a total of 2000 snapshots per run. In contrast to the reproductive runs discussed above where snapshots were collected in a somewhat periodic region, for these runs snapshots are collected starting from the initial condition.

A single POD basis was computed using snapshots from the Mach 0.575 and Mach 0.625 ROM runs. The initial flow from the Mach 0.6 case was used as the base flow and subtracted from both sets of snapshots. This simplifies the ROM creation, but the resulting basis is tailored for the Mach 0.6 case. There are other options for handling the base flow which may produce a more general basis, but there is no clear best approach and it is an area of future research.

Figure 5.13 shows the time pressure time histories for the FOM and two predictive ROMs run with 170 and 750 modes out of 4000 total modes. In both cases, the ROMs appear to reproduce the frequency of the pressure variations at the beginning of the simulations. However, the amplitudes of the oscillations are not captured well and appear to be growing. Eventually the ROMs produce non-physical results, such as negative energies or pressures. Although these results may seem discouraging, they are inconclusive in light of the unexpected behavior observed in the reproductive LSPG ROMs discussed earlier (e.g., non-zero error when $M = K$). As mentioned earlier, issues must be resolved and the reproductive ROMs verified prior to performing more studies of the predictive capabilities of the SPARC ROMs.

5.1.2 Inviscid pulse in uniform base flow

This test case consists of a 2D Gaussian pressure pulse in the domain: $\Omega = [-1.0, 1.0] \times [-1.0, 1.0]$. The purpose of this much simpler problem is to try to identify and resolve the SPARC LSPG ROM non-convergence (as $M \rightarrow K$) and apparent instability issues identified in Section 5.1.1.

Slip wall boundary conditions (i.e. no-penetration, inviscid walls) are applied to the edges of the

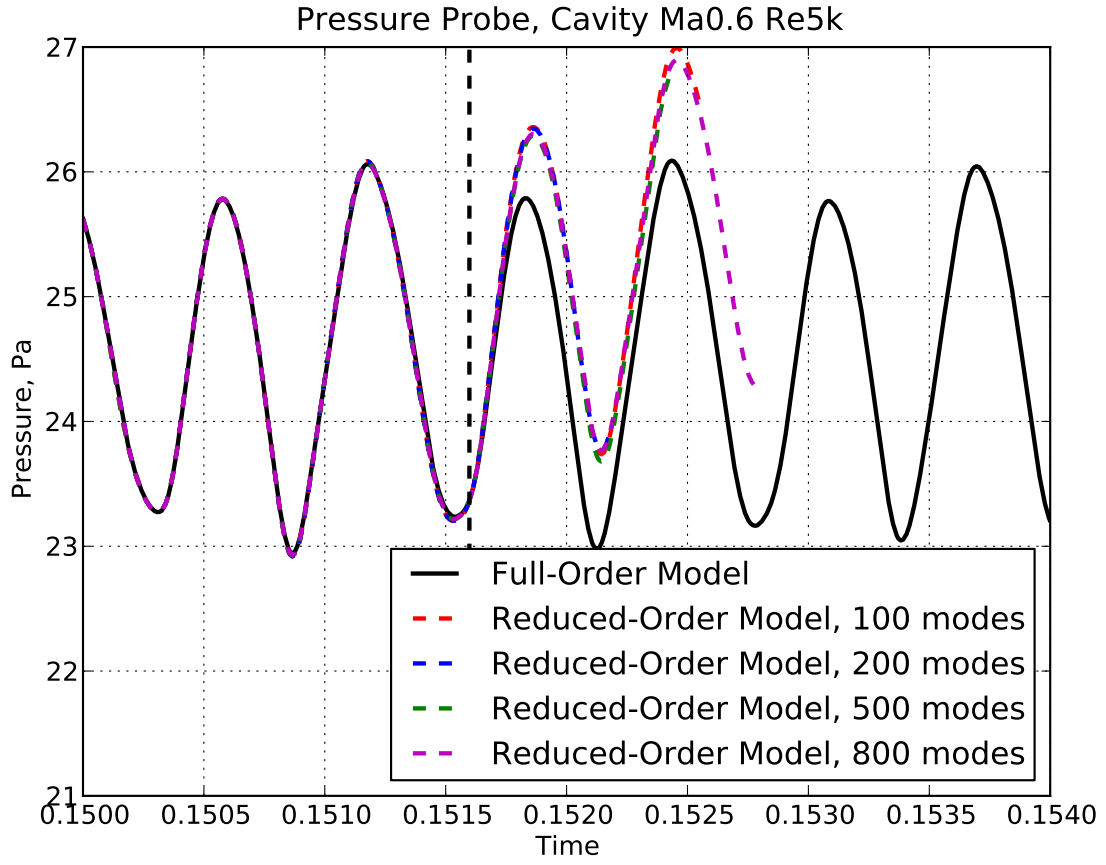


Figure 5.12. The pressure time histories at a point midway up the downstream wall of the cavity for the FOM and ROMs run for an extended time period using a POD basis constructed from 800 snapshots. The dashed vertical line indicates the end of the snapshot collection period. Curves terminating before time 0.154 indicate that a non-physical quantity was computed by the ROM, and the ROM was unable to continue.

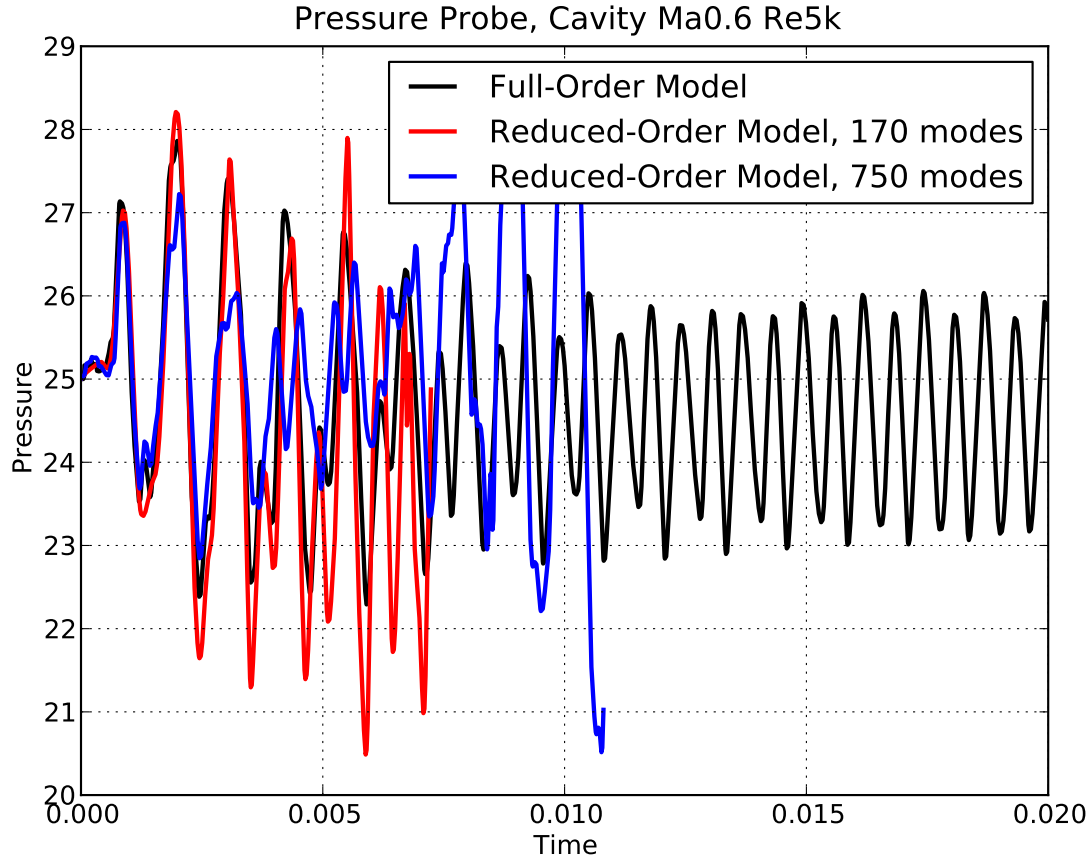


Figure 5.13. The pressure time histories at a point midway up the downstream wall of the cavity for the FOM and ROMs run at Mach 0.6 when the POD basis was constructed using snapshots from Mach 0.575 and Mach 0.625 runs. Curves terminating before time 0.02 indicate that a non-physical quantity was computed by the ROM, and the ROM was unable to continue.

Parameter	Dimensional Value	Non-Dimensional Value
Pressure, \bar{p}	$101325.0 N/m^2$	1.0
Temperature, \bar{T}	$301.24825911 K$	1.0
Density, $\bar{\rho}$	$1.1715554 kg/m^3$	1.4
Velocities, \bar{u}_1 and \bar{u}_2	$0.0 m/s$	0.0
Speed of Sound, \bar{c}	$347.97 m/s$	1.0
Specific Gas Constant, R	$287.097384766765 m^2/(s^2 K)$	0.7142857142857143
γ	1.4	1.4
Time Step, Δt	$1.0e - 5 s$	$3.4797e - 3$

Table 5.2. Parameters used for the inviscid pulse test case.

domain. SPARC is a 3D CFD code, so the computational domain is made to be one cell thick and symmetry (i.e. slip wall) boundary conditions are applied on the faces parallel to the plane.

For the results presented here, the domain is discretized into $40 \times 40 \times 1$ hexahedral cells. This choice of discretization was inherited from previous runs in Sigma CFD [22]. However, the Sigma CFD runs used a higher-order method, the Rai scheme, whereas SPARC is currently only 2nd-order. A mesh convergence study in SPARC is currently underway to determine if this discretization is sufficient.

The test case is initialized by applying disturbances to the uniform base flow of the form:

$$p' = 0.001\gamma\bar{p}e^{-10r}, \quad (5.1)$$

$$T' = 0.0, \quad (5.2)$$

$$\rho' = \frac{p'}{R\bar{T}}, \quad (5.3)$$

$$u'_1 = 0.0, \quad (5.4)$$

$$u'_2 = 0.0, \quad (5.5)$$

where γ is the ratio of specific heats (1.4 for air), R is the specific gas constant, r is the radial distance from the center of the domain, and \bar{p} and \bar{T} are the pressure and temperature of the base flow.

This test case was run both dimensionally and non-dimensionally¹. The purpose of these two sets of runs was to determine whether poor scaling in the (default) dimensional runs can be contributing to the apparent instability and non-convergence properties of the LSPG ROMs in SPARC described in Section 5.1.1. The parameter values used for these runs are given in Table 5.2. In each case, whether the run was dimensional or non-dimensional, the workflow for creating and running a ROM was the same. For the ROM runs in SPARC, the test case was run for 1000 time steps and snapshots of the flow field were recorded every 5 time steps. Additional runs with 4000 and 8000 time steps were performed for the stabilization work in Spirit (see Section 5.2).

¹As mentioned in Section 3.3, SPARC is a dimensional code. However, it is possible to run a non-dimensional problem through a careful selection of the initial conditions and parameters appearing in the equations.

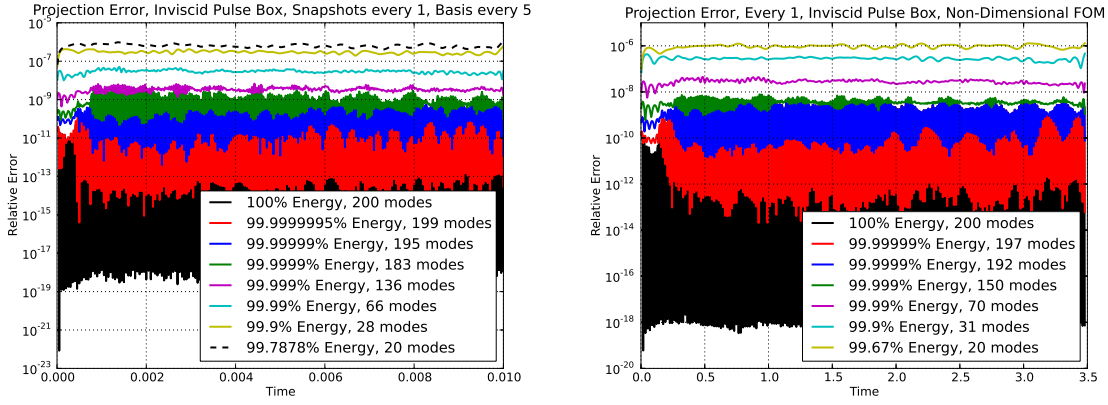


Figure 5.14. The projection errors for the POD basis constructed using the FOM run dimensionally (left) vs. non-dimensionally (right) for the inviscid pulse problem. The POD basis for each case was constructed using every 5 snapshots, but the projection error is computed for every snapshot. This can cause the error to increase substantially for snapshots not in the training set

First, SPARC was run using the standard full-order model implementation for 1000 time steps and snapshots of the flow field were collected every 5 time steps for a total of 200 snapshots. These snapshots were then used to construct a POD basis containing a maximum of 200 modes. For these runs, the flow field associated with the initial pressure pulse was used as the base flow. SPARC was then run using the reduced-order model implementation to try to reproduce the full-order model behavior.

Figure 5.14 shows the projection error for the POD basis computed from the dimensional and non-dimensional FOM runs. The POD basis for each case was constructed using every 5 snapshots. However, the projection error shown here is computed for every snapshot. The projection error is low for snapshots included in the training set, but the error can increase substantially for those snapshots not in the training set. This is evident by the large spread seen for the 200 mode cases and other cases using a large number of modes. Overall, the projection error is low for all cases, meaning that for each case the POD basis is capable of accurately reproducing the snapshots from the FOM.

For the results presented here, LSPG ROMs were run using varying numbers of modes. Figure 5.15 provides some results for both the dimensional and non-dimensional runs. These plots show the generalized coordinate (i.e. coefficient) for one mode, in this case mode 4, for ROMs using 5 to 200 modes. Also shown on these plots is the full-order model solution projected onto the mode of interest, i.e. mode 4. If the time history of the generalized coordinates matches the projected full-order model for all modes, then the ROM will accurately reproduce the FOM behavior.

The plot on the left in Figure 5.15 shows the behavior for the dimensional runs. As can be seen, the 5 and 10 mode ROMs do not accurately replicate the FOM. The 30 mode ROM diverges from the expected behavior and quickly produces NaNs. The 20 and 70 mode ROMs seem to do fairly

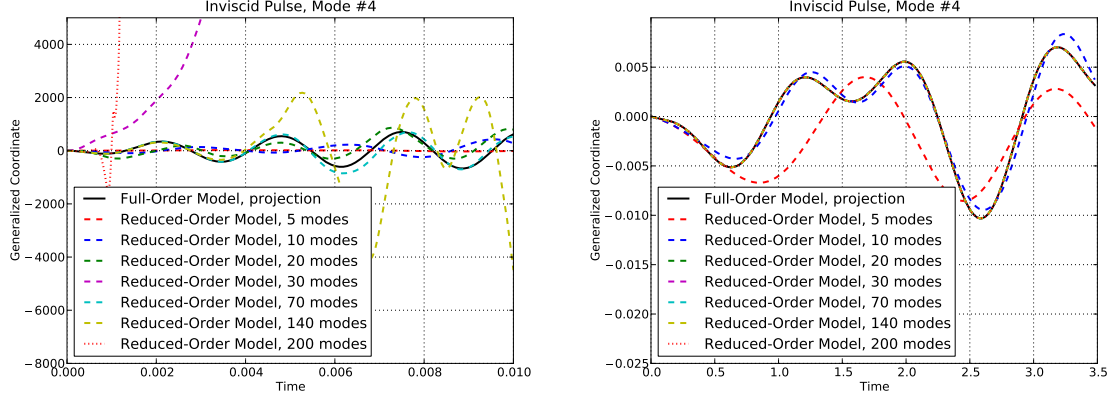


Figure 5.15. Time history of SPARC LSPG ROM generalized coordinates for mode 4 compared to FOM projection for the inviscid pulse problem run dimensionally (left) vs. non-dimensionally (right).

well, but they do not exactly replicate the FOM. The 140 mode ROM seems to exhibit a possible instability, but it remains finite over the simulation time. The 200 mode ROM seems to go unstable very early and quickly produces NaNs. This behavior was unexpected. LSPG ROMs should have the property that adding additional modes does not negatively impact the accuracy of the ROM.

The plot on the right in Figure 5.15 shows the behavior for the non-dimensional runs. The observed behavior for these runs is much more consistent with expectations. The 5 mode ROM is fairly accurate at reproducing the projected FOM solution, and increasing the number of modes to 10 improves the accuracy. Further increases to the number of modes produces ROMs whose behavior is indistinguishable visually from the projected FOM behavior. These results suggest that the unexpected behavior in the dimensional results was at least partially due to poor scaling, as conjectured above; however, in order to ascertain this, additional testing on a variety of problems is required. Such studies, as well as more quantitative evaluations of the accuracy, as planned, but not presented here.

5.2 Numerical evaluation of ROM stabilized via minimal subspace rotation

The results in Section 5.1 suggest that the LSPG ROMs in SPARC could benefit from basis enhancement and stabilization via minimal subspace rotation (Chapter 4), as this approach has been shown to improve the stability and accuracy of low-dimensional ROMs [5, 4]. First, we present some promising results for a variant of the viscous laminar cavity problem described above for which a Galerkin ROM was generated entirely in the *Spirit* code using snapshots from the *Sigma* CFD flow solver (Section 5.2.1). Such ROMs are termed “*Sigma* CFD–*Spirit*” ROMs. Next, in Section 5.2.2, we apply steps (i)–(v) of the stabilization algorithm outlined in Section 4.2 to an

LSPG ROM constructed in SPARC for the inviscid pulse test case considered in Section 5.1.2, and evaluate the performance of the resulting SPARC–Spirit ROM.

5.2.1 Sigma CFD–Spirit Galerkin ROM stabilized via minimal subspace rotation for the viscous laminar cavity

Here, we present some preliminary results for a variant of the viscous laminar cavity problem described above for which a Galerkin ROM was generated entirely in the Spirit code using snapshots from the Sigma CFD flow solver, and stabilized using the minimal subspace rotation approach described in Chapter 4. The purpose of this study is to gauge the viability of the proposed subspace rotation approach for long-time moderate Reynolds number problems similar to those simulated using SPARC.

The domain of interest is an open cavity, discretized using a mesh with 98,408 nodes (see Figure 3 in [4]). At the inflow boundary, a value of the velocity and temperature that is above the free stream values is specified. The flow at the cavity walls is assumed to be adiabatic and to satisfy a no-slip condition. The remaining outflow boundaries are open, and a far-field boundary condition that suppresses the reflection of waves into the computational domain is implemented here. The high-fidelity simulation is initialized by setting the flow in the cavity to have a zero velocity, free stream pressure, and temperature. The region above the cavity is initialized to free stream conditions and the flow is allowed to evolve. The viscosity μ is kept spatially constant and calculated such that a Reynolds number of 5452.1 is achieved. The thermal conductivity κ is also constant, and computed such that the Prandtl number is 0.72. The Mach number of the flow is 0.6.

The high-fidelity solver, Sigma CFD, was initiated with the conditions described above and allowed to run until a statistically stationary flow regime is reached. At this point, a total of $K_{max} = 500$ snapshots were collected from SIGMA CFD, taken every $\Delta t_{snap} = 1 \times 10^{-5}$ seconds. The snapshots were used to construct a POD basis of size 20 modes in the L^2 inner product. This basis captured about 72% of the snapshot energy. Typically, the basis size M would be selected such that the POD basis captures a greater percentage of the snapshot ensemble energy (e.g., $\approx 90\%$ or more). We chose a basis that captures less energy of the snapshot set to highlight the effectiveness of our approach for low-dimensional POD expansions. The predictive power of the stabilized ROM was evaluated by numerically integrating the ROM $10\times$ the duration of the original snapshots.

Figure 5.16 illustrates the performance of a stabilized $M = P = 20$ ROM of the higher Reynolds number cavity problem. In this figure, the modal energy of the DNS, standard, and stabilized ROMs are illustrated. The standard ROM does not go unstable for this problem, but overpredicts the energy of the original DNS snapshots by an order of magnitude.

Figure 5.17 shows the PSD of the predicted pressure fluctuations at the point half way up the downstream wall of the cavity. The PSD of the CFD signal was computed using all available snapshots from $t = 0$ to $t = 67$ where t is non-dimensional. On the other hand, the PSD of the stabilized ROM was computed from the signal $10\times$ past the duration of the original snapshots; i.e. $t = (670 - 67)$ to $t = 670$. The stabilized ROM accurately predicts the chaotic pressure fluctuations

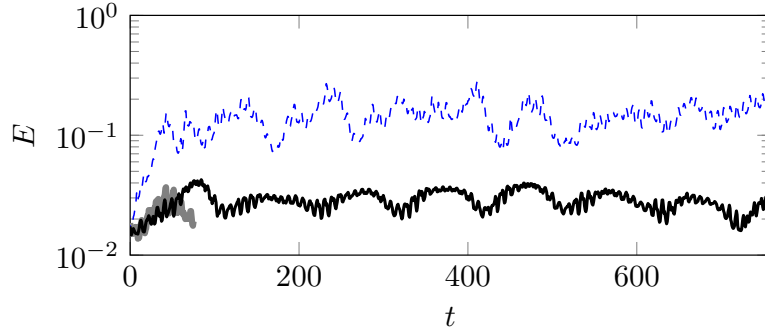


Figure 5.16. Nonlinear model reduction of the cavity at $\text{Re} \approx 5500$. Evolution of modal energy; DNS (thick gray line), standard $M = 20$ ROM (dashed blue line), stabilized $M = P = 20$ ROM (solid black line).

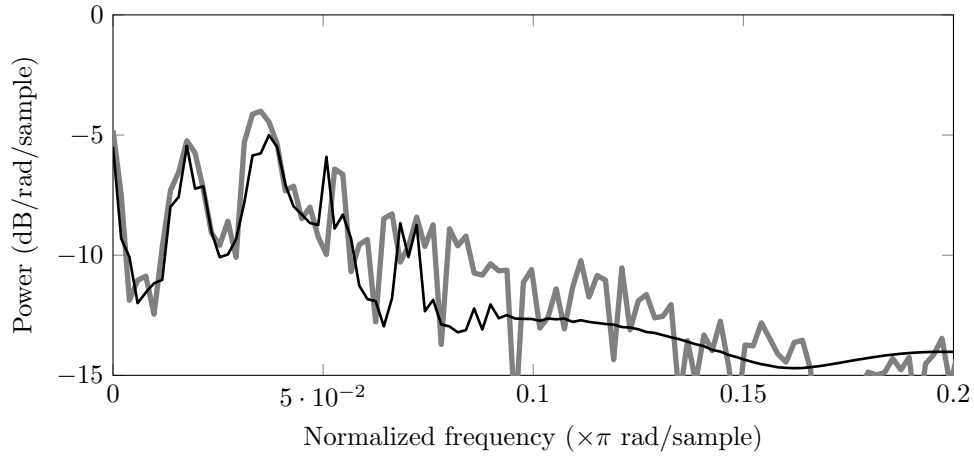


Figure 5.17. PSD of pressure at the point half way up the downstream wall of the cavity for the cavity at $\text{Re} \approx 5500$. DNS (thick gray line), stabilized $M = P = 20$ ROM (black line)

at the selected location.

Figure 5.18 illustrates the Cross Power Spectral Density (CPSD) for pressure fluctuations at two points: half way up the downstream wall of the cavity, and half way up the upstream wall of the cavity. Both the power and phase lag at the fundamental frequency, and the first two super harmonics (normalized frequency ($\times \pi$ rad/sample) $\approx 0.18, 0.35$, and 0.53) are predicted accurately using the stabilized ROM. The phase lag at these three frequencies in Figure 5.18 as predicted by the CFD and the stabilized ROM is identified by red squares and blue triangles, respectively. As expected, the low-dimensional ROM is unable to reproduce the phase lag of low-amplitude frequencies or higher-order super harmonics.

Finally, Figure 5.19 shows a snapshot of the predicted velocity and pressure magnitudes at the final snapshot collection time. Since the flow at this higher Reynolds number is chaotic, the low-dimensional model can not be expected to track the original snapshots exactly. However, the

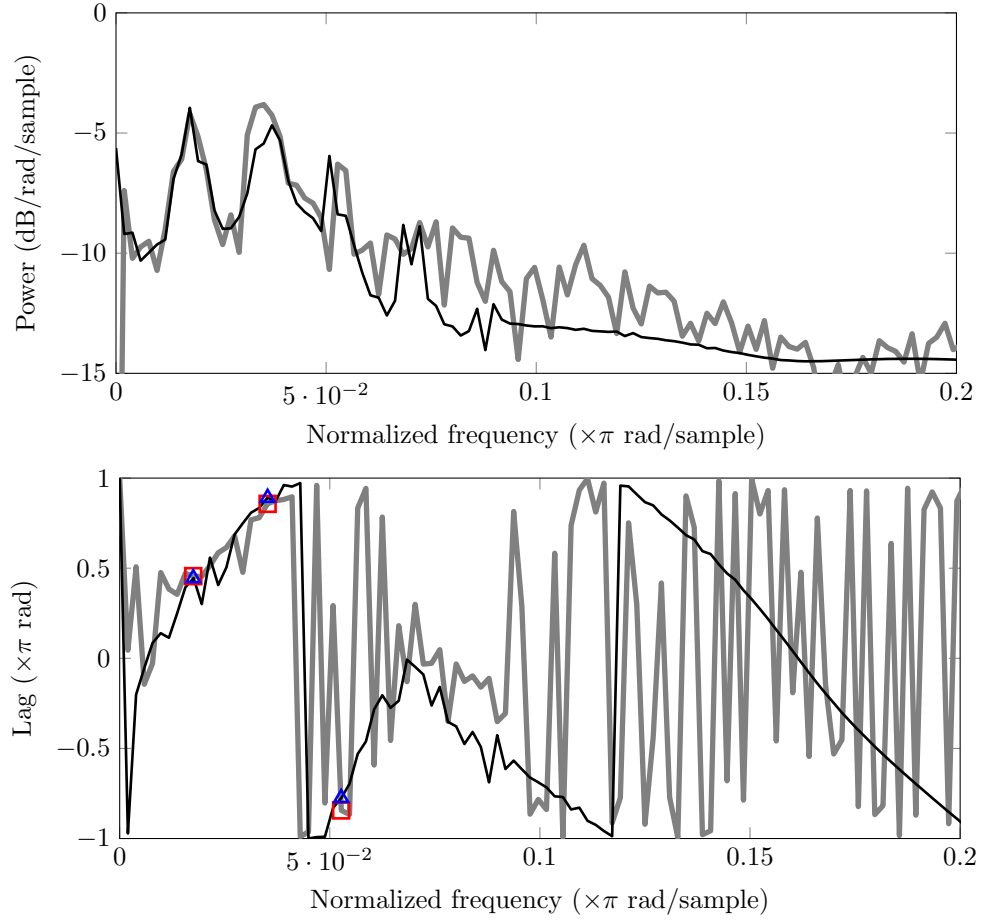


Figure 5.18. CPSD of the pressure between two points inside the of the cavity at $\text{Re} \approx 5500$. DNS (thick gray line), stabilized $M = P = 20$ ROM (black line)

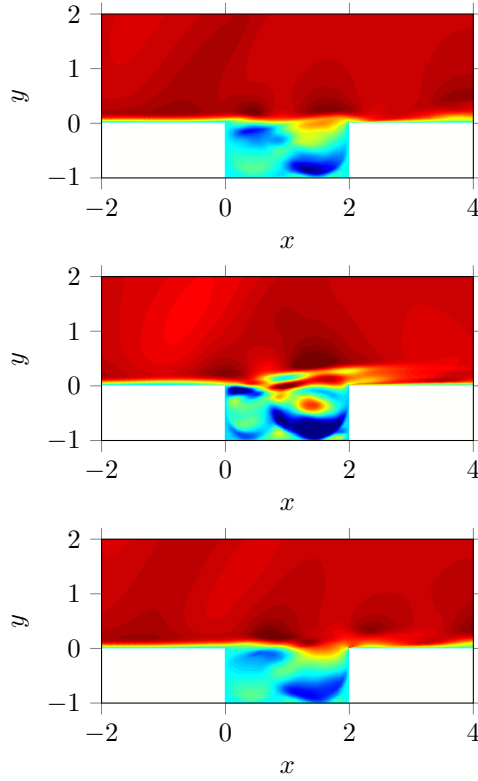


Figure 5.19. Snapshot of the cavity $Re \approx 5500$; contours of u -velocity magnitude at the final snapshot. DNS (top), standard $M = 20$ ROM (middle), and stabilized $M = P = 20$ ROM (bottom)

snapshots demonstrates that the stabilized ROMs faithfully reproduce the large features of the flow. The same cannot be said of the standard ROMs.

5.2.2 SPARC–Spirit LSPG ROM stabilized via minimal subspace rotation for the inviscid pulse in uniform base flow

Having demonstrated the viability of the basis rotation and enhancement technique described in Chapter 4 on a Sigma CFD–Spirit Galerkin ROM, we consider the extension of this method to a SPARC LSPG ROM following the procedure outlined in Section 4.2. The problem considered is the viscous inviscid pulse in a uniform base flow (Section 5.1.2). We have attempted to stabilize and enhance the LSPG ROMs in SPARC for this problem by applying steps (i)–(v) in Section 4.2. That is, we have stabilized a given SPARC basis for this problem using the Spirit code and Algorithm 1 in Chapter 4, and compared the generalized coordinates with the projection of the snapshots onto the stabilized POD modes for the resulting SPARC–Spirit ROMs. Applying the full LSPG stabilization workflow described in Section 4.2, in particular, reading the stabilized basis into SPARC and evaluating the accuracy of the resulting LSPG ROM in SPARC, will be the subject of future work.

As described in [22, 17], implemented within the `Spirit` code are a number of linear as well as nonlinear compressible flow equations. Since the inviscid pulse case considered here is effectively linear, we run `Spirit` in linearized Euler mode. It is straightforward to show that a Galerkin ROM for the linear Euler equations has the form

$$\frac{d\mathbf{a}}{dt} = \mathbf{L}\mathbf{a}. \quad (5.6)$$

To be consistent with the Euler equations, the Galerkin ROM must be marginally stable; that is, the real components of the eigenvalues of \mathbf{L} must vanish. For the specific case of a linear system, the stabilization optimization problem (4.10) takes the form

$$\begin{aligned} & \underset{\mathbf{X} \in \mathcal{V}_{(M+P),M}}{\text{minimize}} && -\text{tr}(\mathbf{X}^T \mathbf{I}_{(M+P) \times M}) \\ & \text{subject to} && \text{Re}(\tilde{\boldsymbol{\lambda}}) = 0, \end{aligned} \quad (5.7)$$

where $\tilde{\boldsymbol{\lambda}}$ are the eigenvalues of $\mathbf{X}^T \mathbf{L} \mathbf{X}$. This optimization problem is solved in `MATLAB` using the active-set algorithm in `fmincon`. Derivatives are approximated using central finite differences and work on incorporating analytical derivatives is ongoing.

Numerical results for an $M = P = 10$ `SPARC-Spirit` ROM stabilized using the minimal subspace rotation approach are presented. The reader can observe by examining Figure 5.20 that the stabilized ROM remains marginally stable for all time t , while the underlying `SPARC-Spirit` ROM exhibits exponential growth.

The decaying amplitudes of the FOM projection coefficients indicates that the FOM solver is dissipative. This is likely due to the 2nd order scheme employed in `SPARC`. Since the marginal stability constraint in (5.7) ($\text{Re}(\tilde{\boldsymbol{\lambda}}) = 0$) is non-dissipative, if the user wishes to use the marginal stability constraint, the FOM discretization must be upgraded to a higher-order, less dissipative scheme. It is possible to generate more accurate ROMs using the existing 2nd-order FOM discretization in `SPARC` by modifying (5.7) to require asymptotic stability (i.e., $\text{Re}(\tilde{\boldsymbol{\lambda}}) < 0$) and using the ODE based objective given in (4.14). Results for this problem are shown in Figure 5.21.

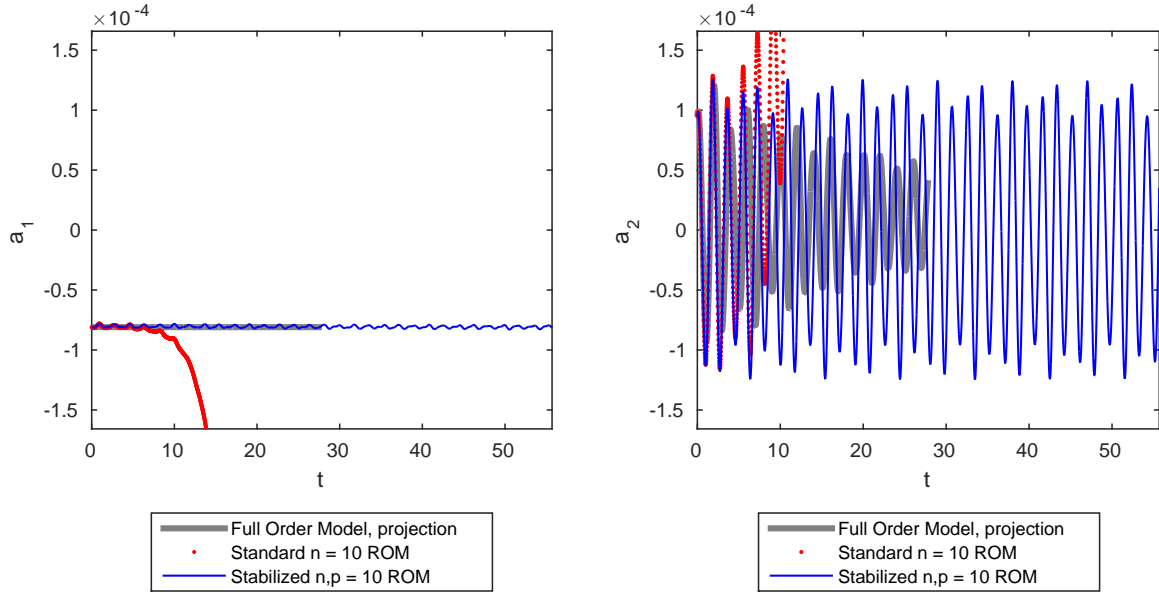


Figure 5.20. Time history of SPARC-Spirit $M = 10$ mode Galerkin ROM generalized coordinates compared to FOM projection for inviscid pulse problem run: marginal stability constraint.

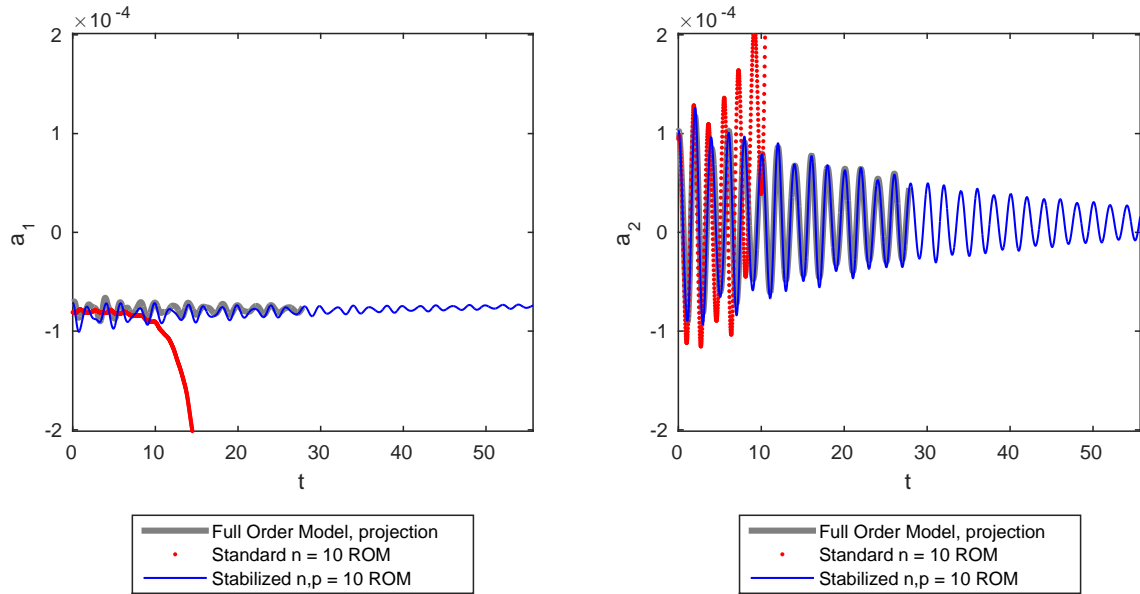


Figure 5.21. Time history of SPARC-Spirit $M = 10$ mode Galerkin ROM generalized coordinates compared to FOM projection for inviscid pulse problem run: asymptotic stability constraint.

Chapter 6

Discussion

This report summarizes our FY16 progress towards enabling uncertainty quantification using model reduction for compressible cavity simulations performed using the SPARC in-house flow solver. In particular, we have described the following:

- our model reduction methodology, which is based on the POD/Galerkin and POD/LSPG methods,
- the implementation of these model reduction methods in SPARC,
- the minimal subspace rotation approach for reduced basis stabilization and enhancement,
- numerical results for reproductive as well as predictive ROMs for several test cases simulated in SPARC, including a test case involving a cavity geometry at a moderate Reynolds number.

Here, we summarize the main takeaways from the numerical studies performed during FY16 and detailed in the previous chapter, as well as describe our plan for subsequent work.

6.1 LSPG ROMs in SPARC

Although we have made progress towards understanding the expected performance of POD/LSPG ROMs for compressible captive cavity flow problems simulated using SPARC, further work is needed to make a definitive conclusion about their viability. First and foremost, we must understand why reproductive full basis ($M = K$) LSPG ROMs in SPARC have non-zero error, a result inconsistent with the theory of LSPG ROMs. Our numerical experiments for the inviscid pulse test case lead us to believe that this issue can be resolved through a simple non-dimensionalization (or equivalent scaling) of the equations in SPARC. A reformulation of the boundary condition treatment for the ROMs in SPARC may also be required. Only after the ROM implementation in SPARC for the reproductive case is corrected and verified does it make sense to evaluate the predictive capabilities of the SPARC LSPG ROMs and complete the implementation of hyper-reduction in SPARC.

6.2 Basis stabilization and enhancement via minimal subspace rotation

We have obtained promising results applying the minimal subspace rotation approach for basis stabilization and enhancement to compressible cavity flow problems in the moderate Reynolds number regime using the `Sigma` CFD high-fidelity flow solver and the `Spirit` ROM code: the stabilized ROMs are stable for long-time simulations and capable of predicting statistical properties (e.g., pressure PSDs) of the flow. Our original formulation of this method was for ROMs constructed for the governing equations in primitive specific volume form. We have formulated an extension of the stabilization and enhancement methodology to LSPG ROMs in `SPARC`, and have put in place most of the steps required. The full workflow has yet to be completed and tested. Since we believe the apparent instability and non-convergence issues encountered in our numerical studies of the LSPG ROMs in `SPARC` can be remedied through non-dimensionalization and/or boundary condition reformulation, the minimal subspace rotation approach may not be as critical as originally believed for reproductive ROMs in which the basis captures a large fraction of the snapshot energy (e.g., 99% or more). However, the method *is* expected to improve stability and accuracy for low-dimensional ROMs. A worthwhile research endeavor in the future is to come up with a further extension of the minimal subspace rotation approach which is embedded within the `SPARC` flow solver and does not require the `Spirit` code, to ensure consistency between the FOM and stabilized ROM systems. For predictive simulations using ROMs, additional research in expanding the span of a reduced basis (stabilized or not) beyond that of the original snapshot set is needed as well. This is because, for predictive problems, the minimal subspace rotation approach will not yield a more accurate result than that obtained using a full-basis ($M = K$) ROM.

6.3 FY17 project plan

Much of the FY17 effort will be focused on resolving the computational difficulties described above by correcting the ROM implementations in `SPARC`. Once these issues are remedied, we will move towards higher Reynolds number, turbulent problems. We will also resume the implementation of hyper-reduction in `SPARC`, which will enable us to achieve large speed-ups in our ROMs. We will continue the extension of the minimal subspace rotation technique for stabilizing and enhancement projection based ROMs to minimal-residual ROMs in `SPARC`.

Also included in the FY17 effort will be the implementation and verification of higher order discretization methods (e.g., the Rai scheme) in the `SPARC` high-fidelity solver. Higher order methods will enable us to use coarser meshes in our numerical examples, and lead to less dissipative FOM and ROM solutions.

At the end of FY17, we will produce a report documenting our captive carriage ROM findings during the year, and (if applicable) recommendations of alternative approaches based on literature search.

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] P. Astrid, S. Weiland, K. Willcox, and T. Backx. Missing point estimation in models described by proper orthogonal decomposition. *IEEE Transactions on Automatic Control*, 53(10):2237–2250, 2008.
- [3] N. Aubry, P. Holmes, J. Lumley, and E. Stone. The dynamics of coherent structures in the wall region of a turbulent boundary layer. *Psychology Today and Tomorrow*, 192:115–173, 1988.
- [4] 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.
- [5] M.J. Balajewicz, E.H. Dowell, and B.R. Noack. Low-dimensional modelling of high-Reynolds-number shear flows incorporating constraints from the Navier-Stokes equation. *J. Fluid Mech.*, 729:285–308, 2013.
- [6] A. Barbagallo, D. Sipp, and P.J. Schmid. Closed-loop control of an open cavity flow using reduced-order models. *J. Fluid Mech.*, 641:1–50, 2009.
- [7] 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.
- [8] M.F. Barone, I. Kalashnikova, M.R. Brake, and D.J. Segalman. Reduced order modeling of fluid/structure interaction. Sandia National Laboratories Report, SAND No. 2009-7189, Sandia National Laboratories, Albuquerque, NM, 2009.
- [9] 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.
- [10] N. Boumal, B. Mishra, P. Absil, and R. Sepulchre. Manopt, a Matlab toolbox for optimization on manifolds. *Journal of Machine Learning Research*, 15:1455–1459, 2014.
- [11] T. Bui-Thanh, K. Willcox, O. Ghattas, and B. van Bloemen Waanders. Goal-oriented, model constrained optimization for reduction of large-scale systems. *J. Comput. Phys.*, 224:880–

896, 2007.

- [12] K. Carlberg, M. Barone, and H. Antil. Galerkin v. least-squares Petrov–Galerkin projection in nonlinear model reduction. *arXiv e-print*, (1504.03749), 2015.
- [13] K. Carlberg, C. Farhat, and C. Bou-Mosleh. Efficient non-linear model reduction via a least-squares Petrov–Galerkin projection and compressive tensor approximations. *International Journal for Numerical Methods in Engineering*, 86(2):155–181, April 2011.
- [14] 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.
- [15] S. Chaturantabut and D.C. Sorensen. Nonlinear model reduction via discrete empirical interpolation. *SIAM J. Sci. Comput.*, 32:2737–2764, 2010.
- [16] R. Everson and L. Sirovich. Karhunen-Loeve procedure for gappy data. *J. Optical Society of America A*, pages 1657–1664, 1995.
- [17] J.A. Fike. Construction of reduced order models for the non-linear Navier-Stokes equations using the proper orthogonal decomposition (POD)/Galerkin method. Sandia National Laboratories Report, SAND No. 2013-7051, Sandia National Laboratories, Albuquerque, NM, 2013.
- [18] S. Gugercin and A.C. Antoulas. A survey of model reduction by balanced truncation and some new results. *Int. J. Control*, 77(8):748–766, 2004.
- [19] 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.
- [20] P. Holmes, J.L. Lumley, and G. Berkooz. *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*. Cambridge University Press, 1996.
- [21] A. Iollo, S. Lanteri, and J. Désidéri. Stability properties of pod–galerkin approximations for the compressible navier–stokes equations. *Theoretical and Computational Fluid Dynamics*, 13(6):377–396, 2000.
- [22] 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.
- [23] I. Kalashnikova and M.F. Barone. On the stability and convergence of a Galerkin reduced order model (ROM) of compressible flow with solid wall and far-field boundary treatment. *J. Comput. Phys. Int. J. Numer. Meth. Engng.*, 83:1345–1375, 2010.

- [24] I. Kalashnikova, M.F. Barone, S. Arunajatesan, and B.G. van Bloemen Waanders. Construction of energy-stable Galerkin reduced order models. Sandia National Laboratories Report, SAND No. 2013-4063, Sandia National Laboratories, Albuquerque, NM, 2013.
- [25] I. Kalashnikova, M.F. Barone, S. Arunajatesan, and B.G. van Bloemen Waanders. Construction of energy-stable projection-based reduced order models. *Appl. Math. Computat.*, pages 569–596, 2014.
- [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] J.L. Lumley. *Stochastic tools in turbulence*. Academic Press: New York, 1971.
- [29] B. Moore. Principal component analysis in linear systems: Controllability, observability, and model reduction. *IEEE Transactions on Automatic Control*, 26(1), 1981.
- [30] 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.
- [31] 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.
- [32] J. Nocedal and S.J. Wright. *Numerical Optimization*. Springer, 1999.
- [33] T. Rapcsák. On minimization on Stiefel manifolds. *European Journal of Operational Research*, 143(2):365–376, 2002.
- [34] M. Rathinam and L.R. Petzold. A new look at proper orthogonal decomposition. *SIAM J. Num. Anal.*, 41(5):1893–1925, 2003.
- [35] D. Rempfer and H.F. Fasel. Dynamics of three-dimensional coherent structures in a flat-plate boundary layer. *Journal of Fluid Mechanics*, 275:257–283, 1994.
- [36] C.W. Rowley. Model reduction for fluids using balanced proper orthogonal decomposition. *Int. J. Bif. Chaos*, 15(3):997–1013, 2005.
- [37] C.W. Rowley, T. Colonius, and A.J. Basu. On self-sustained oscillations in two-dimensional compressible flow over rectangular cavities. *J. Fluid Mech.*, 455:315–346, 2002.
- [38] C.W. Rowley, I. Mezic, S. Baheri, P. Schlatter, and D.S. Henningson. Reduced-order models for flow control: balanced models and Koopman modes. *Seventh IUTAM Symposium on Laminar-Turbulent Transition*, 2009.
- [39] G. Rozza. Reduced basis approximation and error bounds for potential flows in parametrized geometries. *Commun. Comput. Phys.*, 9(1):1–48, 2011.

- [40] A. Salinger, R. Bartlett, A. Bradley, Q. Chen, I. Demeshko, X. Gao, G. Hansen, A. Mota, R. Muller, E. Nielsen, J. Ostien, R. Pawlowski, M. Perego, E. Phipps, W. Sun, and I. Tezaur. Albany: Using agile components to develop a flexible, generic multiphysics analysis code. *Int. J. Multiscale Comput. Engng. (in press)*, 2016.
- [41] P.J. Schmid. Dynamic mode decomposition of numerical and experimental data. *Journal of Fluid Mechanics*, 656:5–28, 2010.
- [42] L. Sirovich. Turbulence and the dynamics of coherent structures, part iii: dynamics and scaling. *Q. Appl. Math.*, 45(3):583–590, 1987.
- [43] E. Stiefel. Richtungsfelder und fernparallelismus in n-dimensionalen mannigfaltigkeiten. *Commentarii Mathematici Helvetici*, 8(1):305–353, 1935.
- [44] K. Veroy and A.T. Patera. Certified real-time solution of the parametrized steady incompressible Navier-Stokes equations: rigorous reduced-bases *a posteriori* error bounds. *J. Num. Meth. Fluids*, 47:773–788, 2005.
- [45] Z. Wang, I. Akhtar, J. Borggaard, and T. Iliescu. Proper orthogonal decomposition closure models for turbulent flows: a numerical comparison. *Computer Methods in Applied Mechanics and Engineering*, 237:10–26, 2012.
- [46] K. Willcox and J. Peraire. Balanced model reduction via the proper orthogonal decomposition. *AIAA Journal*, 40(11):2323–2330, 2002.

DISTRIBUTION:

1	MS 0825	Srini Arunajatesan, 01515
1	MS 0825	Matt Barone, 01515
1	MS 1070	Jeff Fike, 01526
1	MS 9159	Kevin Carlberg, 08959
1	MS 9957	Erin Mussoni, 08253
1	MS 9159	Youngsoo Choi, 08959
1	MS 9159	Jerry McNeish, 08954
1	MS 9159	Gayle Thayer, 08959
1	MS 9159	Liqian Peng, 08959
1	MS 0825	Micah Howard, 01515
1	MS 0845	Miguel Aguilo, 01542
1	MS 9159	Chi Hoang, 08959
1	MS 0899	Technical Library, 9536 (electronic copy)

