arXiv:1504.03749v2 [cs.NA] 8 Dec 2015

Galerkin v. least-squares Petrov–Galerkin projection in nonlinear model reduction

Kevin Carlberg[a,*], Matthew Barone[a,*], Harbir Antil[b,**]

[a] *Sandia National Laboratories*
[b] *George Mason University*

Abstract

Least-squares Petrov–Galerkin (LSPG) model-reduction techniques such as the Gauss–Newton with Approximated Tensors (GNAT) method have shown promise, as they have generated stable, accurate solutions for large-scale turbulent, compressible flow problems where standard Galerkin techniques have failed. However, there has been limited comparative analysis of the two approaches. This is due in part to difficulties arising from the fact that Galerkin techniques perform optimal projection at the time-continuous level, while LSPG techniques do so at the time-discrete level.

This work provides a detailed theoretical and computational comparison of the two techniques for two common classes of time integrators: linear multistep schemes and Runge–Kutta schemes. We present a number of new findings, including conditions under which the LSPG ROM has a time-continuous representation, conditions under which the two techniques are equivalent, and time-discrete error bounds for the two approaches. Perhaps most surprisingly, we demonstrate both theoretically and computationally that decreasing the time step does not necessarily decrease the error for the LSPG ROM; instead, the time step should be 'matched' to the spectral content of the reduced basis. In numerical experiments carried out on a turbulent compressible-flow problem with over one million unknowns, we show that increasing the time step to an intermediate value decreases both the error and the simulation time of the LSPG reduced-order model by an order of magnitude.

Keywords: model reduction, GNAT, least-squares Petrov–Galerkin projection, Galerkin projection, CFD

1. Introduction

While modeling and simulation of parameterized systems has become an essential tool in many industries, the computational cost of executing high-fidelity simulations is infeasibly high for many time-critical applications. For example, real-time scenarios (e.g., model predictive control) require simulations to execute in seconds or minutes, while many-query scenarios (e.g., sampling statistical inversion) can require thousands of simulations corresponding to different parameter instances of the system.

Reduced-order models (ROMs) have been developed to mitigate this computational bottleneck. First, they execute an *offline* stage during which computationally expensive training tasks (e.g., evaluating the high-fidelity model at several points in the parameter space) compute a representative low-dimensional 'trial' basis for the system state. Then, during the inexpensive *online* stage, these methods quickly compute approximate solutions for arbitrary points in the parameter space via projection: they compute solutions in the span of the trial basis while enforcing the high-fidelity-model residual to be orthogonal to a low-dimensional 'test' basis. They also introduce other approximations in the presence of general nonlinearities or non-affine parameter dependence. See Ref. [13] and references within for a survey of current methods.

*7011 East Ave, MS 9159, Livermore, CA 94550. Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under contract DE-AC04-94-AL85000.

**4400 University Drive, MS: 3F2, Exploratory Hall, Room 4201, Fairfax, Virginia 22030.

Email addresses: `ktcarlb@sandia.gov` (Kevin Carlberg), `mbarone@sandia.gov` (Matthew Barone), `hantil@gmu.edu` (Harbir Antil)

URL: `sandia.gov/~ktcarlb` (Kevin Carlberg)

By far the most popular model-reduction technique for nonlinear ordinary differential equations (ODEs) is Galerkin projection [53], wherein the test basis is set to be equal to the trial basis, which is often computed via proper orthogonal decomposition (POD) [35]. Galerkin projection can be considered *continuous optimal*, as the Galerkin-ROM velocity minimizes the ODE (time-continuous) residual in the ℓ^2-norm. In addition, for specialized dynamical systems (e.g., Lagrangian dynamical systems), performing Galerkin projection is necessary to preserve problem structure [39, 22, 23]. However, theoretical analyses—in the form of time-continuous error bounds [45] and stability analysis [31]—as well as numerical experiments have shown that Galerkin projection can lead to significant problems when applied to general nonlinear ODEs: instability [46], inaccurate long-time responses [52, 43], and no guarantee of *a priori* convergence (i.e., adding basis vectors can degrade the solution) [47, Section 5]. In turbulent fluid flows, some of this poor performance can be attributed to the trial basis 'filtering out' small-scale modes essential for energy dissipation.

To address these shortcomings, alternative projection techniques have been developed, particularly in fluid dynamics. These include stabilizing inner products [47, 11, 38]; introducing dissipation via closure models [8, 52, 14, 57, 50] or numerical dissipation [36]; performing nonlinear Galerkin projection based on approximate inertial manifolds [41, 51, 37]; including a pressure-term representation [43, 33]; modifying the POD basis by including many modes (such that dissipative modes are captured), changing the norm [36], enabling adaptivity [14], or including basis functions that resolve a range of scales [9] or respect the attractor's power balance [10]; and performing Petrov–Galerkin projection [28].

Alternatively, a promising new model-reduction methodology eschews Galerkin projection in favor of performing projection at the *fully discrete level*, i.e., after the ODE has been discretized in time [19]. This *discrete-optimal* method—known as least-squares Petrov–Galerkin (LSPG) projection—computes the solution that minimizes the ℓ^2-norm of the (time-discrete) residual arising at each time step; this leads to a notion of *a priori* convergence [17, Theorem 2.2], as adding basis vectors guarantees monotonic decrease in the least-squares objective function. When equipped with gappy POD [27] (a least-squares generalization of—and precursor to—the discrete empirical interpolation method [24]) to approximate the discrete residual as a complexity-reduction mechanism, this approach is known as the Gauss–Newton with Approximated Tensors (GNAT) method [21]. While LSPG projection does not necessarily guarantee *a priori* accuracy and stablility for turbulent, compressible flows, it has been computationally demonstrated to significantly outperform Galerkin projection on such problems [20, 21].

In spite of its promise, theoretical analysis has been limited to developing consistency conditions for snapshot collection [19, 21] and discrete-time error bounds for simple time integrators [21, 3]. In particular, major outstanding questions include: (1) What are time-continuous and time-discrete representations of the Galerkin and LSPG ROMs for broad classes of time integrators? (2) Are there conditions under which the two techniques are equivalent? (3) What discrete-time error bounds are available for the two techniques for broad classes time integrators? Related to the third issue is how parameters (e.g., time step or basis dimension) for the LSPG ROM should be chosen to optimize performance. This work aims to fill this gap by performing a number of detailed theoretical and computational studies that compare Galerkin and LSPG ROMs for the two most important classes of time integrators: linear multistep methods and Runge–Kutta schemes. We summarize the most important theoretical results (which map to the three questions above) as follows:

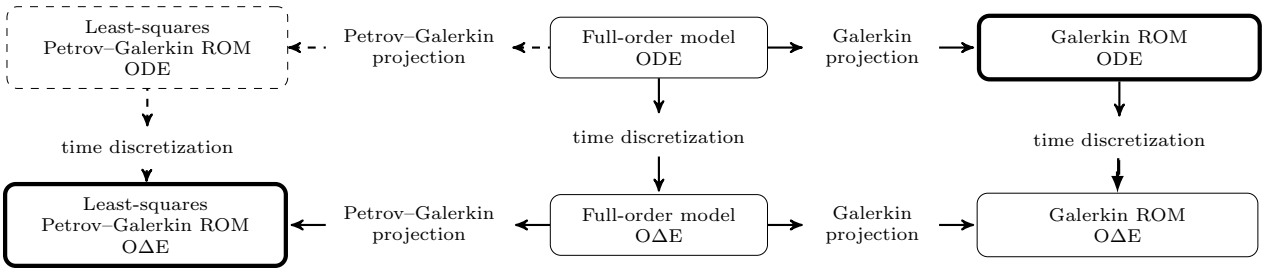


Figure 1: Relationship between Galerkin and discrete-optimal ROMs at the time-continuous (ODE) and time-discrete (OΔE) levels. Bolded outlines imply the ROM associates with a minimum-residual solution at that time-discretization level. Dashed lines imply the relationships are valid under certain conditions (see Theorems 4.2 and 4.3).

1. Continuous and discrete representations
   - Galerkin projection and time discretization are commutative (Theorem 3.4).
   - LSPG ROMs can be derived for Runge–Kutta schemes (Section 4.1).
   - The LSPG ROM has a time-continuous (i.e., ODE) representation under certain conditions (Section 4.2, Figure 1). This ODE depends on the time step Δt.
2. Equivalence conditions
   - Galerkin and LSPG ROMs are equivalent for explicit time integrators (Corollaries 5.1 and 5.2).
   - Galerkin and LSPG ROMs are equivalent in the limit of $\Delta t \to 0$ for implicit time integrators (Theorem 5.3).
   - Galerkin ROMs are discrete optimal for symmetric-positive-definite residual Jacobians (Theorems 5.4–5.6).
3. Error analysis
   - We provide discrete-time error bounds for both Galerkin and LSPG ROMs for linear multistep schemes (Section 6.1).
   - For the backward Euler time integrator, we show that the LSPG ROM yields a lower global state-space 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 (Corollary 6.4).
   - For the backward Euler time integrator, we show that *an intermediate* time step should yield the lowest error bound (Corollary 6.5 and Remark 6.6).
   - For the backward Euler time integrator, we show that a larger basis size leads to a smaller optimal time step for the LSPG ROM (Corollary 6.5).
   - We provide discrete-time error bounds for both Galerkin and LSPG ROMs for Runge–Kutta schemes (Section 6.3).

Figure 1 summarizes time-continuous and time-discrete representations of the two techniques.

In addition to the above theoretical results, we present numerical results for a large-scale compressible fluid-dynamics problem with turbulence modeling characterized by over one million degrees of freedom. These results illustrate the practical significance of the above theoretical results. Critically, we show that employing an intermediate time step for the LSPG ROM can decrease both the error and the simulation time by an order of magnitude, which is a highly non-intuitive result that is of immense practical significance.

The remainder of the paper is organized as follows. Section 2 formulates the full-order model, including its representation at the time-continuous and time-discrete levels. Section 3 presents the Galerkin ROM at the continuous and discrete levels, and Section 4 does so for the LSPG ROM. In particular, we provide conditions under which the LSPG ROM has a time-continuous representation. Section 5 provides conditions under which the Galerkin and LSPG ROMs are equivalent; in particular, equivalence holds for explicit integrators (Section 5.1), in the limit of the time step going to zero for implicit integrators (Section 5.2), and for symmetric-positive-definite residual Jacobians (Section 5.3). Section 6 provides error analysis for Galerkin and LSPG ROMs for linear multistep schemes (Section 6.1), Runge–Kutta schemes (Section 6.3), and a detailed analysis in the case of backward Euler (Section 6.2). Section 7 provides detailed numerical examples that illustrate the practical importance of the analysis. Finally, Section 8 provides conclusions.

In the remainder of this paper, matrices are denoted by capitalized bold letters, vectors by lowercase bold letters, scalars by unbolded letters. The columns of a matrix $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ are denoted by $\boldsymbol{a}_i \in \mathbb{R}^m$, $i \in \mathbb{N}(n)$ with $\mathbb{N}(a) := \{1, \ldots, a\}$ such that $\boldsymbol{A} := [\boldsymbol{a}_1 \ \cdots \ \boldsymbol{a}_n]$. The scalar-valued matrix elements are denoted by $a_{ij} \in \mathbb{R}$ such that $\boldsymbol{a}_j := \begin{bmatrix} a_{1j} & \cdots & a_{mj} \end{bmatrix}^T$, $j \in \mathbb{N}(n)$. A superscript denotes the value of a variable at that time instance, e.g., $\boldsymbol{x}^n$ is the value of $\boldsymbol{x}$ at time $n\Delta t$, where Δt is the time step.

2. Full-order model

We begin by formulating both the time-continuous (ODE) and time-discrete (OΔE) representations of the full-order model (FOM).

2.1. Continuous representation

In this work, we consider the full-order model (FOM) to be an initial value problem characterized by a system of nonlinear ODEs

$$\frac{d\boldsymbol{x}}{dt} = \boldsymbol{f}(\boldsymbol{x}, t), \qquad \boldsymbol{x}(0) = \boldsymbol{x}_0, \tag{2.1}$$

where $\boldsymbol{x} : [0, T] \to \mathbb{R}^N$ denotes the (time-dependent) state, $\boldsymbol{x}_0 \in \mathbb{R}^N$ denotes the initial condition, and $\boldsymbol{f} : \mathbb{R}^N \times [0, T] \to \mathbb{R}^N$ with $(\boldsymbol{\xi}, \tau) \mapsto \boldsymbol{f}(\boldsymbol{\xi}, \tau)$. This ODE can arise, for example, from applying spatial discretization (e.g., finite element, finite volume, or finite difference) to a partial differential equation with time dependence. We note that most model-reduction techniques are applied to parameterized systems wherein the velocity $\boldsymbol{f}$ is also parameter dependent. However, we limit our presentation to unparameterized systems for notational simplicity, as we are interested comparing Galerkin and LSPG ROMs for a given instance of the ODE.

2.2. Discrete representation

A time-discretization method is required to solve Eq. (2.1) numerically. We now characterize the full-order-model OΔE, which is the time-discrete representation of the model, for two classes of time integrators: linear multistep schemes and Runge–Kutta schemes.

2.2.1. Linear multistep schemes

A linear k-step method applied to numerically solve Eq. (2.1) can be expressed as

$$\sum_{j=0}^{k} \alpha_j \boldsymbol{x}^{n-j} = \Delta t \sum_{j=0}^{k} \beta_j \boldsymbol{f}\left(\boldsymbol{x}^{n-j}, t^{n-j}\right), \tag{2.2}$$

where Δt is the time step, the coefficients α_j and β_j define a specific linear multistep scheme, $\alpha_0 \neq 0$, and $\sum_{j=0}^{k} \alpha_j = 0$ is necessary for consistency. In this case, the OΔE is characterized by the following system of algebraic equations to be solved at each time instance $n \in \mathbb{N}(T/\Delta t)$:

$$\boldsymbol{r}^n(\boldsymbol{w}^n) = 0, \tag{2.3}$$

where $\boldsymbol{w}^n \in \mathbb{R}^N$ is the unknown variable and $\boldsymbol{r}^n : \mathbb{R}^N \to \mathbb{R}^N$ denotes the linear multistep residual defined as

$$\boldsymbol{r}^n(\boldsymbol{w}) := \alpha_0 \boldsymbol{w} - \Delta t \beta_0 \boldsymbol{f}(\boldsymbol{w}, t^n) + \sum_{j=1}^{k} \alpha_j \boldsymbol{x}^{n-j} - \Delta t \sum_{j=1}^{k} \beta_j \boldsymbol{f}\left(\boldsymbol{x}^{n-j}, t^{n-j}\right). \tag{2.4}$$

Then, the state can be updated explicitly as

$$\boldsymbol{x}^n = \boldsymbol{w}^n.$$

Hence, the unknown is equal to the state. These methods are implicit if $\beta_0 \neq 0$.

2.2.2. Runge–Kutta schemes

For an s-stage Runge–Kutta scheme, the OΔE is characterized by the following system of algebraic equations to be solved at each time step:

$$\boldsymbol{r}_i^n(\boldsymbol{w}_1^n, \ldots, \boldsymbol{w}_s^n) = 0, \quad i \in \mathbb{N}(s). \tag{2.5}$$

Here, the Runge–Kutta residual is defined as

$$\boldsymbol{r}_i^n(\boldsymbol{w}_1, \ldots, \boldsymbol{w}_s) := \boldsymbol{w}_i - \boldsymbol{f}(\boldsymbol{x}^{n-1} + \Delta t \sum_{j=1}^{s} a_{ij} \boldsymbol{w}_j, t^{n-1} + c_i \Delta t), \quad i \in \mathbb{N}(s) \tag{2.6}$$

and the state is explicitly updated as

$$\boldsymbol{x}^n = \boldsymbol{x}^{n-1} + \Delta t \sum_{i=1}^{s} b_i \boldsymbol{w}_i^n. \tag{2.7}$$

Here, the unknowns $\boldsymbol{w}_i^n$ correspond to the velocity $d\boldsymbol{x}/dt$ at times $t^{n-1} + c_i \Delta t$, $i \in \mathbb{N}(s)$, and the coefficients b_i, c_i, and a_{ij} define a specific Runge–Kutta scheme. These methods are explicit if $a_{ij} = 0$, $\forall j \geq i$ and are diagonally implicit if $a_{ij} = 0$, $\forall j > i$.

3. Galerkin ROM

This section provides the time-continuous and time-discrete representations of the Galerkin ROM, as well as key results related to optimality and commutativity of projection and time discretization.

3.1. Continuous representation

Galerkin-projection reduced-order models compute an approximate solution $\tilde{\boldsymbol{x}} \approx \boldsymbol{x}$ with $\tilde{\boldsymbol{x}} \in \mathbb{R}^N$ to Eq. (2.1) by introducing two approximations. First, they restrict the approximate solution to lie in a low-dimensional affine trial subspace $\boldsymbol{x}_0 + \operatorname{range}(\boldsymbol{\Phi})$, where $\boldsymbol{\Phi} \in \mathbb{R}^{N \times p}$ with $\boldsymbol{\Phi}^T \boldsymbol{\Phi} = I$ denotes the given reduced basis (in matrix form) of dimension $p \ll N$. This basis can be computed by a variety of techniques, e.g., eigenmode analysis, POD [35], or the reduced-basis method [44, 48, 56, 42, 55]. Then, the approximate solution can be expressed as

$$\tilde{\boldsymbol{x}}(t) = \boldsymbol{x}_0 + \boldsymbol{\Phi} \hat{\boldsymbol{x}}(t), \tag{3.1}$$

where $\hat{\boldsymbol{x}} : [0, T] \to \mathbb{R}^p$ denotes the (time-dependent) generalized coordinates of the approximate solution. Second, these methods substitute $\boldsymbol{x} \leftarrow \tilde{\boldsymbol{x}}$ into Eq. (2.1) and enforce the ODE residual to be orthogonal to $\operatorname{range}(\boldsymbol{\Phi})$, which results in a low-dimensional system of nonlinear ODEs

$$\frac{d\hat{\boldsymbol{x}}}{dt} = \boldsymbol{\Phi}^T \boldsymbol{f}(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}, t), \qquad \hat{\boldsymbol{x}}(0) = 0. \tag{3.2}$$

Remark 3.1. In order to obtain computational efficiency, it is necessary to reduce the computational complexity of repeatedly computing matrix–vector products of the form $\boldsymbol{\Phi}^T \boldsymbol{f}$. This can be done using a variety of methods, e.g., collocation [7, 49, 40], gappy POD [27, 16, 7, 19, 21], the discrete empirical interpolation method (DEIM) [12, 24, 32, 26, 6], reduced-order quadrature [5], finite-element subassembly methods [4, 29], or reduced-basis-sparsification techniques [23]. However, in this work we limit ourselves to comparatively analyzing different projection techniques. For this reason, we do not perform additional analysis for such complexity-reduction mechanisms; this is the subject of follow-on work.

We now restate the well-known result that Galerkin projection leads to a notion of minimum-residual optimality at the continuous level. This is reflected in the top-right box of Figure 1, where the bolded outline indicates minimum-residual optimality.

Theorem 3.2 (Galerkin: continuous optimality). *If the reduced basis is orthogonal, i.e.,* $\boldsymbol{\Phi}^T \boldsymbol{\Phi} = I$*, then the Galerkin ROM* (3.1)–(3.2) *is continuous optimal in the sense that the approximated velocity minimizes the* ℓ^2*-norm of the FOM ODE residual* (2.1) *over* $\operatorname{range}(\boldsymbol{\Phi})$*, i.e.,*

$$\frac{d\tilde{\boldsymbol{x}}}{dt}(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}, t) = \arg \min_{\boldsymbol{v} \in \operatorname{range}(\boldsymbol{\Phi})} \|\boldsymbol{v} - \boldsymbol{f}(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}, t)\|_2^2. \tag{3.3}$$

Proof. Because $\frac{d\tilde{\boldsymbol{x}}}{dt} = \boldsymbol{\Phi}\frac{d\hat{\boldsymbol{x}}}{dt}$, problem (3.3) can be written as

$$\frac{d\hat{\boldsymbol{x}}}{dt}(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}, t) = \arg\min_{\hat{\boldsymbol{v}}\in\mathbb{R}^p} g\left(\hat{\boldsymbol{v}}\right), \tag{3.4}$$

where $g\left(\hat{\boldsymbol{v}}\right) := \|\boldsymbol{\Phi}\hat{\boldsymbol{v}} - \boldsymbol{f}(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}, t)\|_2^2$. We now assess whether Eq. (3.4) holds, i.e., whether $\frac{d\hat{\boldsymbol{x}}}{dt}$ as defined by Eq. (3.2) is the minimizer of g.

The function g can be expressed as $g\left(\hat{\boldsymbol{v}}\right) = \hat{\boldsymbol{v}}^T\boldsymbol{\Phi}^T\boldsymbol{\Phi}\hat{\boldsymbol{v}} - 2\hat{\boldsymbol{v}}^T\boldsymbol{\Phi}^T\boldsymbol{f}(\boldsymbol{x}_0+\boldsymbol{\Phi}\hat{\boldsymbol{x}}, t) + \boldsymbol{f}(\boldsymbol{x}_0+\boldsymbol{\Phi}\hat{\boldsymbol{x}}, t)^T\boldsymbol{f}(\boldsymbol{x}_0+\boldsymbol{\Phi}\hat{\boldsymbol{x}}, t)$. Due to the strict convexity of the function g, the global minimizer $\hat{\boldsymbol{v}}^\star$ is equal to the stationary point of g, i.e., $\hat{\boldsymbol{v}}^\star$ satisfies

$$0 = \frac{dg}{d\hat{\boldsymbol{v}}}\left(\hat{\boldsymbol{v}}^\star\right) = 2\boldsymbol{\Phi}^T\boldsymbol{\Phi}\hat{\boldsymbol{v}}^\star - 2\boldsymbol{\Phi}^T\boldsymbol{f}(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}, t) \tag{3.5}$$

$$\hat{\boldsymbol{v}}^\star = \boldsymbol{\Phi}^T\boldsymbol{f}(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}, t), \tag{3.6}$$

where orthogonality of $\boldsymbol{\Phi}$ has been used. Comparing Eqs. (3.6) and (3.2) shows $\frac{d\hat{\boldsymbol{x}}}{dt}(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}, t) = \hat{\boldsymbol{v}}^\star$, which is the desired result. □

Remark 3.3 (Continuous *a priori* convergence of the Galerkin ROM). Due to optimality property (3.3), the Galerkin ROM can be considered *a priori* convergent (in the sense introduced in Ref. [17, Theorem 2.2]) at the continuous level. In particular, adding vectors to the trial basis—which expands the trial subspace range $(\boldsymbol{\Phi})$—results in a monotonic decrease in the minimum-residual objective function in problem (3.3).

Thus, Galerkin ROMs exhibit desirable properties (i.e., minimum-residual optimality and *a priori* convergence) at the time-continuous level. We now derive a time-discrete representation for the Galerkin ROM, noting that these properties are lost at the time-discrete level.

3.2. Discrete representation

As before, a time-discretization method is needed to numerically solve Eq. (3.2). We now characterize the OΔE for the Galerkin ROM.

3.2.1. Linear multistep schemes

A linear k-step method applied to numerically solve Eq. (3.2) can be expressed as

$$\sum_{j=0}^{k}\alpha_j\hat{\boldsymbol{x}}^{n-j} = \Delta t\sum_{j=0}^{k}\beta_j\boldsymbol{\Phi}^T\boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}^{n-j}, t^{n-j}\right).$$

Here, the OΔE is characterized by the following system of algebraic equations to be solved at each time step:

$$\hat{\boldsymbol{r}}^n\left(\hat{\boldsymbol{w}}^n\right) = 0. \tag{3.7}$$

Here, the discrete residual corresponds to

$$\hat{\boldsymbol{r}}^n\left(\hat{\boldsymbol{w}}\right) := \alpha_0\hat{\boldsymbol{w}} - \Delta t\beta_0\boldsymbol{\Phi}^T\boldsymbol{f}(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{w}}, t^n) + \sum_{j=1}^{k}\alpha_j\hat{\boldsymbol{x}}^{n-j} - \Delta t\sum_{j=1}^{k}\beta_j\boldsymbol{\Phi}^T\boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}^{n-j}, t^{n-j}\right) \tag{3.8}$$

and the generalized state is explicitly updated as

$$\hat{\boldsymbol{x}}^n = \hat{\boldsymbol{w}}^n.$$

3.2.2. Runge–Kutta schemes

Applying an s-stage Runge–Kutta method to solve Eq. (3.2) leads to an OΔE characterized by the following system of algebraic equations to be solved at each time step:

$$\hat{\boldsymbol{r}}_i^n\left(\hat{\boldsymbol{w}}_1^n,\ldots,\hat{\boldsymbol{w}}_s^n\right)=0,\quad i\in\mathbb{N}(s). \tag{3.9}$$

Here, discrete the residual is defined as

$$\hat{\boldsymbol{r}}_i^n\left(\hat{\boldsymbol{w}}_1,\ldots,\hat{\boldsymbol{w}}_s\right):=\hat{\boldsymbol{w}}_i-\boldsymbol{\Phi}^T\boldsymbol{f}(\boldsymbol{x}_0+\boldsymbol{\Phi}\hat{\boldsymbol{x}}^{n-1}+\Delta t\sum_{j=1}^{s}a_{ij}\boldsymbol{\Phi}\hat{\boldsymbol{w}}_j,t^{n-1}+c_i\Delta t),\quad i\in\mathbb{N}(s) \tag{3.10}$$

and the generalized state is computed explicitly as

$$\hat{\boldsymbol{x}}^n=\hat{\boldsymbol{x}}^{n-1}+\Delta t\sum_{i=1}^{s}b_i\hat{\boldsymbol{w}}_i^n. \tag{3.11}$$

Note that the Galerkin-ROM solution satisfying Eqs. (3.7) or (3.7) does not in general associate with the solution to an optimization problem; therefore, the optimality property the method exhibits at the continuous level has been lost at the discrete level. We now show that Galerkin projection and time discretization are commutative; this implies that Galerkin ROMs can be analyzed, implemented, and interpreted equivalently at both the time-discrete and time-continuous levels. This corresponds to the rightmost part of Figure 1.

Theorem 3.4 (Galerkin: commutativity of projection and time discretization). *Performing a Galerkin projection on the governing ODE and subsequently applying time discretization yields the same model as first applying time discretization on the governing ODE and subsequently performing Galerkin projection.*

Proof. Linear multistep schemes. Eq. (3.7) was derived by performing Galerkin projection on the continuous representation of the FOM and subsequently applying time discretization. If instead we apply Galerkin projection to the discrete representation of the FOM in Eq. (2.3), set $\boldsymbol{w}=\boldsymbol{x}_0+\boldsymbol{\Phi}\hat{\boldsymbol{w}}$ and $\boldsymbol{x}^i=\boldsymbol{x}_0+\boldsymbol{\Phi}\hat{\boldsymbol{x}}^i$, $i\in\mathbb{N}(n)$, and use $\sum_{j=1}^k\alpha_j=0$ and $\boldsymbol{\Phi}^T\boldsymbol{\Phi}=I$, we obtain the following OΔE to be solved at each time step: $\boldsymbol{\Phi}^T\boldsymbol{r}^n\left(\boldsymbol{x}_0+\boldsymbol{\Phi}\hat{\boldsymbol{w}}\right)=0$. Comparing the definition of the linear multistep residual (2.4) with Eq. (3.8) reveals

$$\hat{\boldsymbol{r}}^n\left(\hat{\boldsymbol{w}}\right)=\boldsymbol{\Phi}^T\boldsymbol{r}^n\left(\boldsymbol{x}_0+\boldsymbol{\Phi}\hat{\boldsymbol{w}}\right), \tag{3.12}$$

which shows that the same discrete equations $\hat{\boldsymbol{r}}^n\left(\hat{\boldsymbol{w}}\right)=0$ are obtained at each time step regardless of the ordering of time discretization and Galerkin projection.
Runge–Kutta schemes. Eq. (3.9) was derived by performing Galerkin projection on the continuous FOM representation and then applying time discretization. If instead we apply Galerkin projection to the discrete FOM representation in Eq. (2.5), set $\boldsymbol{x}^{n-1}=\boldsymbol{x}_0+\boldsymbol{\Phi}\hat{\boldsymbol{x}}^{n-1}$, $\boldsymbol{w}_i=\boldsymbol{\Phi}\hat{\boldsymbol{w}}_i$, $i\in\mathbb{N}(s)$, and use $\boldsymbol{\Phi}^T\boldsymbol{\Phi}=I$, we obtain the following OΔE to be solved at each time step: $\boldsymbol{\Phi}^T\boldsymbol{r}_i^n\left(\boldsymbol{\Phi}\hat{\boldsymbol{w}}_1,\ldots,\boldsymbol{\Phi}\hat{\boldsymbol{w}}_s\right)=0,\ i\in\mathbb{N}(s)$. Comparing the definition of the Runge–Kutta residual (2.6) with Eq. (3.10) reveals

$$\hat{\boldsymbol{r}}_i^n\left(\hat{\boldsymbol{w}}_1,\ldots,\hat{\boldsymbol{w}}_s\right)=\boldsymbol{\Phi}^T\boldsymbol{r}_i^n\left(\boldsymbol{\Phi}\hat{\boldsymbol{w}}_1,\ldots,\boldsymbol{\Phi}\hat{\boldsymbol{w}}_s\right),\quad i\in\mathbb{N}(s), \tag{3.13}$$

which shows that the same discrete equations $\hat{\boldsymbol{r}}_i^n\left(\hat{\boldsymbol{w}}_1,\ldots,\hat{\boldsymbol{w}}_s\right)=0$, $i\in\mathbb{N}(s)$ are obtained at each time step regardless of the ordering of time discretization and Galerkin projection. □

4. Least-squares Petrov–Galerkin ROM

Rather than performing minimum-residual optimal projection on the full-order model ODE (i.e., at the continuous level), this can be executed on the full-order model OΔE (i.e., at the discrete level). Doing so enables *discrete optimality*, which contrasts with the continuous optimality exhibited by Galerkin projection. In particular, we consider optimal projections that minimize the discrete residual(s) (in some weighted ℓ^2-norm) arising at each time instance.

We note that other residual-minimizing approaches have been developed in the case of linear [17] and nonlinear [40] steady-state problems, and space–time solutions [25]. In addition, a recently developed approach [1] has suggested L^1 minimization of the residual arising at each time instance for hyperbolic problems.

4.1. Discrete representation

We begin by developing the time-discrete representation for the LSPG ROM for both linear multistep schemes and Runge–Kutta schemes. The latter is a novel contribution, as previous work has derived discrete-optimal LSPG ROMs only for linear multistep schemes [19, 21]. Optimality of this approach corresponds to the bolded bottom-left box of Figure 1.

4.1.1. Linear multistep schemes

As before with Galerkin projection, discrete-optimal ROMs compute solutions using two approximations. First, they restrict the approximate solution to lie in the same low-dimensional affine trial subspace $\tilde{\boldsymbol{x}} \in \boldsymbol{x}_0 + \text{range}\left(\boldsymbol{\Phi}\right)$ as Galerkin methods; thus, the approximate solution can be written according to Eq. (3.1). In the case of linear multistep schemes, the unknown at time step n is simply the state, i.e., $\boldsymbol{w}^n = \boldsymbol{x}^n$, which implies that $\tilde{\boldsymbol{w}}^n = \boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{w}}^n$. Second, the discrete-optimal ROM substitutes $\boldsymbol{w}^n \leftarrow \tilde{\boldsymbol{w}}^n$ into the OΔE (2.3) and solves a minimization problem to ensure the approximate solution is optimal in a minimum-residual sense at the discrete level:

$$\tilde{\boldsymbol{w}}^n = \arg \min_{\boldsymbol{z} \in \boldsymbol{x}_0 + \text{range}(\boldsymbol{\Phi})} \left\| \boldsymbol{A}\left(\boldsymbol{z}\right) \boldsymbol{r}^n\left(\boldsymbol{z}\right) \right\|_2^2 \tag{4.1}$$

or equivalently

$$\hat{\boldsymbol{w}}^n = \arg \min_{\hat{\boldsymbol{z}} \in \mathbb{R}^p} \left\| \boldsymbol{A}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{z}}\right) \boldsymbol{r}^n\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{z}}\right) \right\|_2^2. \tag{4.2}$$

Here, $\boldsymbol{A} \in \mathbb{R}^{z \times N}$ with $z \leq N$ is a weighting matrix that enables the definition of a weighted (semi)norm. Examples of such reduced-order models include the LSPG method presented in Refs. [19, 21, 40] ($\boldsymbol{A} = \boldsymbol{I}$), LSPG with collocation ($\boldsymbol{A} = \boldsymbol{P}$ with $\boldsymbol{P}$ consisting of selected rows of the identity matrix) [40], and the related GNAT method [19, 21] ($\boldsymbol{A} = \left(\boldsymbol{P}\boldsymbol{\Phi}_r\right)^+ \boldsymbol{P}$ with $\boldsymbol{\Phi}_r$ a basis for the residual and the superscript $+$ denoting the Moore–Penrose pseudoinverse).

Note that the solution to Eq. (4.2) corresponds to a stationary point of the objective function in Eq. (4.2), i.e., it satisfies

$$\boldsymbol{\Psi}^n (\hat{\boldsymbol{w}}^n)^T \boldsymbol{r}^n\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{w}}^n\right) = 0, \tag{4.3}$$

where the entries of $\boldsymbol{\Psi}^n \in \mathbb{R}^{N \times p}$ are

$$\begin{aligned} \psi_{ij}^n(\hat{\boldsymbol{w}}) =& a_{mi}(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{w}}) \frac{\partial a_{ml}(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{w}})}{\partial w_k} \phi_{kj} r_\ell^n(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{w}}) + \\ & a_{mi}(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{w}}) a_{ml}(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{w}}) \frac{\partial r_\ell^n}{\partial w_k}(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{w}}) \phi_{kj}, \quad i \in \mathbb{N}(N),\ j \in \mathbb{N}(p), \end{aligned} \tag{4.4}$$

where a repeated index implies summation. Because Eq. (4.3) corresponds to a Petrov–Galerkin projection with trial subspace range $\left(\boldsymbol{\Phi}\right)$ and test subspace range $\left(\boldsymbol{\Psi}\right)$, the discrete-optimal projection can be referred to as a least-squares Petrov–Galerkin projection [21, 19].

4.1.2. Runge–Kutta schemes

LSPG ROMs for Runge–Kutta schemes also approximate the solution according to Eq. (3.1). However, because the unknown at time step n and stage i is the velocity at an intermediate time point, i.e., $\boldsymbol{w}_i^n = \dot{\boldsymbol{x}}\left(t^{n-1} + c_i \Delta t\right)$ for $i \in \mathbb{N}(s)$, we have $\tilde{\boldsymbol{w}}_i^n = \boldsymbol{\Phi}\dot{\hat{\boldsymbol{x}}}\left(t^{n-1} + c_i \Delta t\right)$ for this case. Then, these techniques substitute $\boldsymbol{w}^n \leftarrow \tilde{\boldsymbol{w}}^n$ into the OΔE (2.5) and solve the following minimum-residual problem:

$$(\tilde{\boldsymbol{w}}_1^n, \ldots, \tilde{\boldsymbol{w}}_s^n) = \arg \min_{(\boldsymbol{z}_1, \ldots, \boldsymbol{z}_1) \in \text{range}(\boldsymbol{\Phi})^s} \sum_{i=1}^{s} \left\| \boldsymbol{A}_i\left(\boldsymbol{z}_1, \ldots, \boldsymbol{z}_s\right) \boldsymbol{r}_i^n\left(\boldsymbol{z}_1, \ldots, \boldsymbol{z}_s\right) \right\|_2^2 \tag{4.5}$$

or equivalently

$$\left(\hat{\boldsymbol{w}}_1^n, \ldots, \hat{\boldsymbol{w}}_s^n\right) = \arg \min_{(\hat{\boldsymbol{z}}_1, \ldots, \hat{\boldsymbol{z}}_s) \in \mathbb{R}^{p \times s}} \sum_{i=1}^{s} \left\| \boldsymbol{A}_i\left(\boldsymbol{\Phi}\hat{\boldsymbol{z}}_1, \ldots, \boldsymbol{\Phi}\hat{\boldsymbol{z}}_s\right) \boldsymbol{r}_i^n\left(\boldsymbol{\Phi}\hat{\boldsymbol{z}}_1, \ldots, \boldsymbol{\Phi}\hat{\boldsymbol{z}}_s\right) \right\|_2^2. \tag{4.6}$$

Here, $\boldsymbol{A}_i \in \mathbb{R}^{z\times N}$, $i \in \mathbb{N}(s)$ with $z \leq N$ are weighting matrices. As before, the solution to Eq. (4.6) corresponds to a stationary point of the objective function, i.e., it satisfies

$$\sum_{j=1}^{s} \boldsymbol{\Psi}_{ij}^n(\hat{\boldsymbol{w}}_1^n, \ldots, \hat{\boldsymbol{w}}_s^n)^T \boldsymbol{r}_j^n \left(\boldsymbol{\Phi}\hat{\boldsymbol{w}}_1^n, \ldots, \boldsymbol{\Phi}\hat{\boldsymbol{w}}_s^n\right) = 0, \quad i = 1, \ldots, s, \tag{4.7}$$

where entries of the test bases $\boldsymbol{\Psi}_{ij}^n \in \mathbb{R}^{N\times p}$, $i, j \in \mathbb{N}(s)$ are

$$\begin{aligned}\left[\boldsymbol{\Psi}_{ij}^n\right]_{k\ell}(\hat{\boldsymbol{w}}_1, \ldots, \hat{\boldsymbol{w}}_s) =& [\boldsymbol{A}_i]_{uk}(\boldsymbol{\Phi}\hat{\boldsymbol{w}}_1, \ldots, \boldsymbol{\Phi}\hat{\boldsymbol{w}}_s)\frac{\partial[\boldsymbol{A}_i]_{um}(\boldsymbol{\Phi}\hat{\boldsymbol{w}}_1, \ldots, \boldsymbol{\Phi}\hat{\boldsymbol{w}}_s)}{\partial[\boldsymbol{w}_j]_n}\phi_{n\ell}[\boldsymbol{r}_i^n]_m(\boldsymbol{\Phi}\hat{\boldsymbol{w}}_1, \ldots, \boldsymbol{\Phi}\hat{\boldsymbol{w}}_s) + \\ & [\boldsymbol{A}_i]_{uk}(\boldsymbol{\Phi}\hat{\boldsymbol{w}}_1, \ldots, \boldsymbol{\Phi}\hat{\boldsymbol{w}}_s)[\boldsymbol{A}_i]_{um}(\boldsymbol{\Phi}\hat{\boldsymbol{w}}_1, \ldots, \boldsymbol{\Phi}\hat{\boldsymbol{w}}_s)\frac{\partial[\boldsymbol{r}_i^n]_m}{\partial[\boldsymbol{w}_j]_n}(\boldsymbol{\Phi}\hat{\boldsymbol{w}}_1, \ldots, \boldsymbol{\Phi}\hat{\boldsymbol{w}}_s)\phi_{n\ell},\end{aligned} \tag{4.8}$$

where $[\cdot]_{ij}$ denotes entry (i, j) of the argument. This again leads to a least-squares Petrov–Galerkin interpretation for the discrete-optimal ROM.

In the explicit or diagonally implicit cases, we can consider an alternative notion of discrete optimality. Explicit Runge–Kutta schemes are characterized by $a_{ij} = 0$, $\forall j \geq i$, while diagonally implicit Runge–Kutta (DIRK) schemes [2] are characterized by $a_{ij} = 0$, $\forall j > i$. In these cases, solutions $\boldsymbol{w}_i^n$, $i \in \mathbb{N}(s)$ can be computed sequentially, i.e.,

$$\boldsymbol{q}_i^n\left(\boldsymbol{w}_i^n\right) = 0, \quad i = 1, \ldots, s$$

with

$$\boldsymbol{q}_i^n(\boldsymbol{w}) := \boldsymbol{w} - \boldsymbol{f}(\boldsymbol{x}^{n-1} + \Delta t a_{ii}\boldsymbol{w} + \Delta t \sum_{j=1}^{i-1} a_{ij}\boldsymbol{w}_j^n, t^{n-1} + c_i\Delta t), \quad i = 1, \ldots, s. \tag{4.9}$$

We can then formulate the following sequence of optimization problems to compute discrete minimum-residual approximations:

$$\tilde{\boldsymbol{w}}_i^n = \arg\min_{\boldsymbol{z}\in\text{range}(\boldsymbol{\Phi})} \|\boldsymbol{A}_i(\boldsymbol{z})\boldsymbol{q}_i^n(\boldsymbol{z})\|_2^2, \quad i = 1, \ldots, s, \tag{4.10}$$

or equivalently

$$\hat{\boldsymbol{w}}_i^n = \arg\min_{\hat{\boldsymbol{z}}\in\mathbb{R}^p} \|\boldsymbol{A}_i(\boldsymbol{\Phi}\hat{\boldsymbol{z}})\boldsymbol{q}_i^n(\boldsymbol{\Phi}\hat{\boldsymbol{z}})\|_2^2, \quad i = 1, \ldots, s. \tag{4.11}$$

Here, the associated Petrov–Galerkin projection is

$$\boldsymbol{\Psi}_i^n(\hat{\boldsymbol{w}}_i^n)^T\boldsymbol{q}_i^n(\boldsymbol{\Phi}\hat{\boldsymbol{w}}_i^n) = 0, \quad i = 1, \ldots, s, \tag{4.12}$$

with test-basis entries of

$$[\boldsymbol{\Psi}_i^n]_{jk}(\hat{\boldsymbol{w}}) = [\boldsymbol{A}_i]_{uj}(\boldsymbol{\Phi}\hat{\boldsymbol{w}})\frac{\partial[\boldsymbol{A}_i]_{u\ell}(\boldsymbol{\Phi}\hat{\boldsymbol{w}})}{\partial w_m}\phi_{mk}[\boldsymbol{q}_i^n]_\ell(\boldsymbol{\Phi}\hat{\boldsymbol{w}}) + [\boldsymbol{A}_i]_{uj}(\boldsymbol{\Phi}\hat{\boldsymbol{w}})[\boldsymbol{A}_i]_{um}(\boldsymbol{\Phi}\hat{\boldsymbol{w}})\frac{\partial[\boldsymbol{q}_i^n]_m(\boldsymbol{\Phi}\hat{\boldsymbol{w}})}{\partial w_\ell}\phi_{\ell k}, \tag{4.13}$$

Note that in the explicit case, the LSPG ROM generally requires solving a system of nonlinear equations at each Runge–Kutta stage. Because $\partial\boldsymbol{q}_i^n/\partial\boldsymbol{w} = \boldsymbol{I}$, the system of equations is linear if $\boldsymbol{A}_i$ are constant matrices, and only an explicit solution update is required if $\boldsymbol{A}_i = \boldsymbol{I}$ and $\boldsymbol{\Phi}^T\boldsymbol{\Phi} = \boldsymbol{I}$.

Remark 4.1 (Discrete *a priori* convergence of the LSPG ROM). Due to optimality property (4.1), the LSPG ROM can be considered *a priori* convergent (in the sense introduced in Ref. [17, Theorem 2.2]) at the discrete level for linear multistep schemes, as adding vectors to the trial basis—which expands the trial subspace range $(\boldsymbol{\Phi})$—results in a monotonic decrease in the minimum-residual objective function in problem (4.1). This result also holds for LSPG ROMs applied to Runge–Kutta schemes, as the computed solutions satisfy alternative optimality properties in the implicit (4.5) and explicit (4.10) cases.

We now see the critical tradeoff between the Galerkin and LSPG ROMs: Galerkin ROMs exhibit continuous (minimum-residual) optimality, while LSPG ROMs exhibit discrete optimality. Without further analysis, it is unclear which of these attributes is preferable. Numerical experiments (Section 7) and supporting error analysis (Section 6) will highlight the benefits of discrete optimality over continuous optimality in practice.

4.2. Continuous representation

Because the LSPG ROM introduces approximations at the discrete level, it is unclear whether it can be interpreted at the continuous level. In fact, it has not previously been shown that a continuous representation of the LSPG ROM even exists. We now show that an ODE representation of the LSPG ROM does indeed exist for both linear multistep schemes and Runge–Kutta schemes under certain conditions; however, the ODE depends on the time step used to define the LSPG ROM. This associates with the top-left section of the relationship diagram in Figure 1.

Theorem 4.2 (LSPG ROM continuous representation: linear multistep schemes). *The LSPG ROM for linear multistep integrators is equivalent to applying a Petrov–Galerkin projection to the ODE with test basis (in matrix form)*

$$\boldsymbol{\Psi}(\hat{\boldsymbol{x}},t) = \boldsymbol{A}^T\boldsymbol{A}\left(\alpha_0\boldsymbol{I} - \Delta t\beta_0\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{\xi}}(\boldsymbol{x}_0+\boldsymbol{\Phi}\hat{\boldsymbol{x}},t)\right)\boldsymbol{\Phi} \tag{4.14}$$

and subsequently applying time integration with a linear multistep scheme with time step Δt if $\boldsymbol{A}$ is a constant matrix and (at least) one of the following conditions holds:

1. *$\beta_j = 0$, $j \geq 1$ (e.g., a single-step method),*
2. *the velocity $\boldsymbol{f}$ is linear in the state, or*
3. *$\beta_0 = 0$ (i.e., explicit schemes).*

Proof. Applying Petrov–Galerkin projection to the full-order model ODE (2.1) using a trial subspace $\boldsymbol{x}_0 + \operatorname{range}(\boldsymbol{\Phi})$ and test subspace $\operatorname{range}(\boldsymbol{\Psi})$ yields the following ODE

$$\boldsymbol{\Psi}(\hat{\boldsymbol{x}},t)^T\boldsymbol{\Phi}\frac{d\hat{\boldsymbol{x}}}{dt} = \boldsymbol{\Psi}(\hat{\boldsymbol{x}},t)^T\boldsymbol{f}(\boldsymbol{x}_0+\boldsymbol{\Phi}\hat{\boldsymbol{x}},t), \qquad \hat{\boldsymbol{x}}(0) = 0, \tag{4.15}$$

which can be written in standard form as

$$\frac{d\hat{\boldsymbol{x}}}{dt} = \left(\boldsymbol{\Psi}(\hat{\boldsymbol{x}},t)^T\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Psi}(\hat{\boldsymbol{x}},t)^T\boldsymbol{f}(\boldsymbol{x}_0+\boldsymbol{\Phi}\hat{\boldsymbol{x}},t), \qquad \hat{\boldsymbol{x}}(0) = 0. \tag{4.16}$$

<u>Case 1</u> Applying a linear multistep time integrator with the stated assumption of $\beta_j = 0$, $j \geq 1$ to numerically solve Eq. (4.16) results in the following discrete equations to be solved at each time instance:

$$\alpha_0\hat{\boldsymbol{y}}^n - \Delta t\beta_0\left(\boldsymbol{\Psi}(\hat{\boldsymbol{y}}^n,t^n)^T\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Psi}(\hat{\boldsymbol{y}}^n,t^n)^T\boldsymbol{f}(\boldsymbol{x}_0+\boldsymbol{\Phi}\hat{\boldsymbol{y}}^n,t^n) + \sum_{j=1}^{k}\alpha_j\hat{\boldsymbol{x}}^{n-j} = 0. \tag{4.17}$$

Pre-multiplying by $\boldsymbol{\Psi}(\hat{\boldsymbol{y}}^n,t^n)^T\boldsymbol{\Phi}$ yields discrete equations $\hat{\boldsymbol{r}}^n\left(\hat{\boldsymbol{y}}^n\right) = 0$ with residual

$$\hat{\boldsymbol{r}}^n\left(\hat{\boldsymbol{w}}\right) := \alpha_0\boldsymbol{\Psi}(\hat{\boldsymbol{w}},t^n)^T\boldsymbol{\Phi}\hat{\boldsymbol{w}} - \Delta t\beta_0\boldsymbol{\Psi}(\hat{\boldsymbol{w}},t^n)^T\boldsymbol{f}(\boldsymbol{x}_0+\boldsymbol{\Phi}\hat{\boldsymbol{w}},t^n) + \sum_{j=1}^{k}\alpha_j\boldsymbol{\Psi}(\hat{\boldsymbol{w}},t^n)^T\boldsymbol{\Phi}\hat{\boldsymbol{x}}^{n-j}. \tag{4.18}$$

Comparing Eqs. (4.18) and (2.4) reveals $\hat{\boldsymbol{r}}^n\left(\hat{\boldsymbol{w}}\right) = \boldsymbol{\Psi}(\hat{\boldsymbol{w}},t^n)^T\boldsymbol{r}^n\left(\boldsymbol{x}_0+\boldsymbol{\Phi}\hat{\boldsymbol{w}}\right)$ and so the solution $\hat{\boldsymbol{y}}^n$ satisfies

$$\boldsymbol{\Psi}(\hat{\boldsymbol{y}}^n,t^n)^T\boldsymbol{r}^n\left(\boldsymbol{x}_0+\boldsymbol{\Phi}\hat{\boldsymbol{y}}^n\right) = 0. \tag{4.19}$$

Under the stated assumptions, we have $\partial\boldsymbol{r}^n/\partial\boldsymbol{w}(\boldsymbol{x}) = \alpha_0\boldsymbol{I} - \Delta t\beta_0\frac{\partial\boldsymbol{f}}{\partial\boldsymbol{\xi}}(\boldsymbol{x},t^n)$ and so the LSPG test basis $\boldsymbol{\Psi}^n$ defined in Eq. (4.4) is equal to the test basis in Eq. (4.14) evaluated at time instance n, i.e., $\boldsymbol{\Psi}^n(\hat{\boldsymbol{w}}) = \boldsymbol{\Psi}(\hat{\boldsymbol{w}},t^n)$. Therefore, the solution $\hat{\boldsymbol{w}}^n$ to the LSPG OΔE (4.3) satisfies

$$\boldsymbol{\Psi}(\hat{\boldsymbol{w}}^n,t^n)^T\boldsymbol{r}^n\left(\boldsymbol{x}_0+\boldsymbol{\Phi}\hat{\boldsymbol{w}}^n\right) = 0. \tag{4.20}$$

This shows that $\hat{\boldsymbol{w}}^n = \hat{\boldsymbol{y}}^n$, i.e., the solutions to the LSPG OΔE and the OΔE obtained after applying Petrov–Galerkin projection with test basis $\boldsymbol{\Psi}(\boldsymbol{x},t)$ defined by Eq. (4.14) to the full-order model ODE and

subsequently applying time integration are equivalent under the stated assumptions, which is the desired result.

Case 2 In this case, the test basis is independent of the state, i.e.,

$$\boldsymbol{\Psi}(t) = \boldsymbol{A}^T\boldsymbol{A}\left(\alpha_0\boldsymbol{I} - \Delta t\beta_0\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{\xi}}(\cdot, t)\right)\boldsymbol{\Phi}. \tag{4.21}$$

Applying a linear multistep time integrator to solve Eq. (4.16) and subsequently pre-multiplying by the constant matrix $\boldsymbol{\Psi}(t^n)^T\boldsymbol{\Phi}$ yields the following discrete equations arising at each time step

$$\hat{\boldsymbol{r}}^n\left(\hat{\boldsymbol{y}}^n\right) = 0, \tag{4.22}$$

where the residual is defined as

$$\begin{aligned}\hat{\boldsymbol{r}}^n\left(\hat{\boldsymbol{w}}\right) :=& \alpha_0\boldsymbol{\Psi}(t^n)^T\boldsymbol{\Phi}\hat{\boldsymbol{w}} - \Delta t\beta_0\boldsymbol{\Psi}(t^n)^T\boldsymbol{f}(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{w}}, t^n) + \sum_{j=1}^{k}\alpha_j\boldsymbol{\Psi}(t^n)^T\boldsymbol{\Phi}\hat{\boldsymbol{x}}^{n-j} - \\ & \Delta t\sum_{j=1}^{k}\beta_j\boldsymbol{\Psi}(t^n)^T\boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}^{n-j}, t^{n-j}\right).\end{aligned} \tag{4.23}$$

Comparing Eqs. (4.23) and (2.4) reveals $\hat{\boldsymbol{r}}^n\left(\hat{\boldsymbol{w}}\right) = \boldsymbol{\Psi}(t^n)^T\boldsymbol{r}^n\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{w}}\right)$ and so the solution $\hat{\boldsymbol{y}}^n$ satisfies

$$\boldsymbol{\Psi}(t^n)^T\boldsymbol{r}^n\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{y}}^n\right) = 0. \tag{4.24}$$

Under these assumptions, we have $\partial\boldsymbol{r}^n/\partial\boldsymbol{w} = \alpha_0\boldsymbol{I} - \Delta t\beta_0\partial\boldsymbol{f}/\partial\boldsymbol{\xi}(\cdot, t^n)$ and so the LSPG test basis $\boldsymbol{\Psi}^n$ defined in Eq. (4.4) is equal to the test basis in Eq. (4.21) at time instance n, i.e., $\boldsymbol{\Psi}^n(\hat{\boldsymbol{w}}) = \boldsymbol{\Psi}(t^n)$. Therefore, the LSPG O$\Delta$E (4.3) can be expressed as

$$\boldsymbol{\Psi}(t^n)^T\boldsymbol{r}^n\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{w}}^n\right) = 0. \tag{4.25}$$

This shows that $\hat{\boldsymbol{w}}^n = \hat{\boldsymbol{y}}^n$, i.e., the solutions to the LSPG OΔE and the OΔE obtained after applying Petrov–Galerkin projection with test basis $\boldsymbol{\Psi}(t)$ defined by Eq. (4.14) to the full-order model ODE and subsequently applying time integration are equivalent under the stated assumptions.

Case 3 The assumption $\beta_0 = 0$ results in a constant test basis

$$\boldsymbol{\Psi} = \alpha_0\boldsymbol{A}^T\boldsymbol{A}\boldsymbol{\Phi}. \tag{4.26}$$

Applying a linear multistep time integrator to solve Eq. (4.16) and subsequently pre-multiplying by the constant matrix $\boldsymbol{\Psi}^T\boldsymbol{\Phi}$ yields

$$\hat{\boldsymbol{r}}^n\left(\hat{\boldsymbol{y}}^n\right) = 0, \tag{4.27}$$

which is to be solved at each time step with a residual defined as

$$\hat{\boldsymbol{r}}^n\left(\hat{\boldsymbol{w}}\right) := \alpha_0\boldsymbol{\Psi}^T\boldsymbol{\Phi}\hat{\boldsymbol{w}} - \Delta t\beta_0\boldsymbol{\Psi}^T\boldsymbol{f}(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{w}}, t^n) + \sum_{j=1}^{k}\alpha_j\boldsymbol{\Psi}^T\boldsymbol{\Phi}\hat{\boldsymbol{x}}^{n-j} - \Delta t\sum_{j=1}^{k}\beta_j\boldsymbol{\Psi}^T\boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}^{n-j}, t^{n-j}\right). \tag{4.28}$$

As in Case 2, this leads to $\hat{\boldsymbol{r}}^n\left(\hat{\boldsymbol{w}}\right) = \boldsymbol{\Psi}^T\boldsymbol{r}^n\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{w}}\right)$. Because $\frac{\partial\boldsymbol{r}^n}{\partial\boldsymbol{w}}(\boldsymbol{x}) = \alpha_0\boldsymbol{I}$, we also again have $\boldsymbol{\Psi}^n(\hat{\boldsymbol{w}}) = \boldsymbol{\Psi}$. This leads to the desired result, as the OΔEs for the LSPG ROM and the ROM obtained after applying Petrov–Galerkin projection with test basis $\boldsymbol{\Psi}$ to the full-order model ODE and subsequently applying time integration both satisfy $\boldsymbol{\Psi}^T\boldsymbol{r}^n(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{w}}^n) = 0$ under the stated assumptions. □

We now provide conditions under which the LSPG ROM for Runge–Kutta schemes can be expressed as an ODE.

Theorem 4.3 (LSPG ROM continuous representation: Runge–Kutta schemes). *The LSPG ROM for Runge–Kutta integrators is equivalent to applying a Petrov–Galerkin projection to the ODE with test basis (in matrix form)*

$$\boldsymbol{\Psi}(\hat{\boldsymbol{x}}, t) = \boldsymbol{A}^T\boldsymbol{A}\left(\boldsymbol{I} - \Delta ta\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{\xi}}(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}, t)\right)\boldsymbol{\Phi} \tag{4.29}$$

and subsequently applying time integration if $\boldsymbol{A}_i = \boldsymbol{A}$, $\forall i$ are constant matrices and the integrator is a singly diagonally implicit Runge–Kutta (SDIRK) scheme, i.e., $a_{ij} = 0$, $\forall j > i$ and $a_{ii} = a$, $\forall i$.[1]

Proof. Applying Petrov–Galerkin projection to Eq. (2.1) using a trial subspace $\boldsymbol{x}_0 + \text{range}\,(\boldsymbol{\Phi})$ and test subspace $\text{range}\,(\boldsymbol{\Psi})$ yields the following ODE (in standard form)

$$\frac{d\hat{\boldsymbol{x}}}{dt} = \left(\boldsymbol{\Psi}(\hat{\boldsymbol{x}}, t)^T \boldsymbol{\Phi}\right)^{-1} \boldsymbol{\Psi}(\hat{\boldsymbol{x}}, t)^T \boldsymbol{f}(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}, t), \qquad \hat{\boldsymbol{x}}(0) = 0. \tag{4.30}$$

Applying an SDIRK time integrator to numerically solve Eq. (4.30) results in the following sequence of discrete equations to be solved at each time step:

$$\begin{aligned} \hat{\boldsymbol{y}}_i^n - [(\boldsymbol{\Psi}(\hat{\boldsymbol{x}}^{n-1} + \Delta t \sum_{j=1}^{i} a_{ij} \hat{\boldsymbol{y}}_j^n, t^{n-1} + c_i \Delta t)^T \boldsymbol{\Phi}]^{-1} \boldsymbol{\Psi}(\hat{\boldsymbol{x}}^{n-1} + \Delta t \sum_{j=1}^{i} a_{ij} \hat{\boldsymbol{y}}_j^n, t^{n-1} + c_i \Delta t)^T \\ \boldsymbol{f}(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}^{n-1} + \Delta t \sum_{j=1}^{i} a_{ij} \boldsymbol{\Phi} \hat{\boldsymbol{y}}_j^n, t^{n-1} + c_i \Delta t) = 0, \quad i = 1, \ldots, s. \end{aligned} \tag{4.31}$$

Pre-multiplying by $\boldsymbol{\Psi}(\hat{\boldsymbol{x}}^{n-1} + \Delta t \sum_{j=1}^{i} a_{ij} \hat{\boldsymbol{y}}_j, t^{n-1} + c_i \Delta t)^T \boldsymbol{\Phi}$ yields the following discrete equations

$$\hat{\boldsymbol{q}}_i^n \left(\hat{\boldsymbol{y}}_i^n\right) = 0, \quad i = 1, \ldots, s$$

with residual

$$\begin{aligned} \hat{\boldsymbol{q}}_i^n(\hat{\boldsymbol{w}}) := & \boldsymbol{\Psi}(\hat{\boldsymbol{x}}^{n-1} + \Delta t a \hat{\boldsymbol{w}} + \Delta t \sum_{j=1}^{i-1} a_{ij} \hat{\boldsymbol{y}}_j^n, t^{n-1} + c_i \Delta t)^T \cdot \\ & \left[\boldsymbol{\Phi}\hat{\boldsymbol{w}} - \boldsymbol{f}(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}^{n-1} + \Delta t a \hat{\boldsymbol{w}} + \Delta t \sum_{j=1}^{i-1} a_{ij} \boldsymbol{\Phi} \hat{\boldsymbol{y}}_j^n, t^{n-1} + c_i \Delta t) \right]. \end{aligned} \tag{4.32}$$

Comparing Eqs. (4.32) and (4.9) reveals

$$\hat{\boldsymbol{q}}_i^n(\hat{\boldsymbol{w}}) = \boldsymbol{\Psi}(\hat{\boldsymbol{x}}^{n-1} + \Delta t a \hat{\boldsymbol{w}} + \Delta t \sum_{j=1}^{i-1} a_{ij} \hat{\boldsymbol{y}}_j^n, t^{n-1} + c_i \Delta t)^T \boldsymbol{q}_i^n \left(\boldsymbol{\Phi}\hat{\boldsymbol{w}}\right), \quad i = 1, \ldots, s$$

such that the solutions $\hat{\boldsymbol{y}}_i^n$ satisfy

$$\boldsymbol{\Psi}(\hat{\boldsymbol{x}}^{n-1} + \Delta t \sum_{j=1}^{i} a_{ij} \hat{\boldsymbol{y}}_j^n, t^{n-1} + c_i \Delta t)^T \boldsymbol{q}_i^n \left(\hat{\boldsymbol{y}}_i^n\right) = 0, \quad i = 1, \ldots, s. \tag{4.33}$$

Now, under the stated assumptions, we have

$$\frac{\partial \boldsymbol{q}_i^n}{\partial \boldsymbol{w}}(\boldsymbol{u}) = \boldsymbol{I} - \Delta t a \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{\xi}}(\boldsymbol{x}^{n-1} + \Delta t a \boldsymbol{u} + \Delta t \sum_{j=1}^{i-1} a_{ij} \boldsymbol{w}_j^n, t^{n-1} + c_i \Delta t)$$

such that the LSPG test basis $\boldsymbol{\Psi}_i^n$ defined in Eq. (4.13) is related to the test basis in Eq. (4.29) as follows:

$$\boldsymbol{\Psi}_i^n(\hat{\boldsymbol{w}}) = \boldsymbol{\Psi}(\hat{\boldsymbol{x}}^{n-1} + \Delta t a \hat{\boldsymbol{w}} + \Delta t \sum_{j=1}^{i-1} a_{ij} \hat{\boldsymbol{w}}_j^n, t^{n-1} + c_i \Delta t).$$

[1] Note that summation on repeated indicies is not implied.

Therefore, the solutions $\hat{\boldsymbol{w}}_i^n$ to the LSPG OΔE (4.12) satisfy

$$\boldsymbol{\Psi}(\hat{\boldsymbol{x}}^{n-1} + \Delta t \sum_{j=1}^{i} a_{ij} \hat{\boldsymbol{w}}_j^n, t^{n-1} + c_i \Delta t)^T \boldsymbol{q}_i^n \left(\hat{\boldsymbol{w}}_i^n\right) = 0, \quad i = 1, \ldots, s. \tag{4.34}$$

Comparing Eqs. (4.33) and (4.34) shows that the $\hat{\boldsymbol{w}}_i^n = \hat{\boldsymbol{y}}_i^n$, $i \in \mathbb{N}(s)$, i.e., the solutions to the LSPG OΔE and the OΔE obtained after applying Petrov–Galerkin projection with test basis $\boldsymbol{\Psi}(\boldsymbol{x}, t)$ defined by Eq. (4.29) to the full-order model ODE and subsequently applying time integration are equivalent under the stated assumptions, which is the desired result. □

We now show that the LSPG ROM has a time-continuous representation for all explicit and single-state Runge–Kutta schemes, which include the widely used forward Euler, backward Euler, and implicit midpoint schemes.

Corollary 4.4 (LSPG ROM continuous representation: explicit Runge–Kutta). *The LSPG ROM for Runge–Kutta integrators is equivalent to applying a Petrov–Galerkin projection to the ODE with test basis (in matrix form)*

$$\boldsymbol{\Psi}(\hat{\boldsymbol{x}}, t) = \boldsymbol{A}^T \boldsymbol{A} \boldsymbol{\Phi}$$

and subsequently applying time integration if $\boldsymbol{A}_i = \boldsymbol{A}$, $\forall i$ are constant matrices and an explicit Runge–Kutta scheme is employed.

Proof. Explicit Runge–Kutta schemes are characterized by $a_{ij} = 0$, $j \geq i$ and so they satisfy the conditions Theorem 4.3 with $a = 0$. □

Corollary 4.5 (LSPG ROM continuous representation: single-stage Runge–Kutta). *The LSPG ROM for Runge–Kutta integrators is equivalent to applying a Petrov–Galerkin projection to the ODE with test basis (in matrix form)*

$$\boldsymbol{\Psi}(\hat{\boldsymbol{x}}, t) = \boldsymbol{A}^T \boldsymbol{A} \left(\boldsymbol{I} - \Delta t a_{11} \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{\xi}} (\boldsymbol{x}_0 + \boldsymbol{\Phi} \hat{\boldsymbol{x}}, t) \right) \boldsymbol{\Phi}$$

and subsequently applying time integration if $\boldsymbol{A}_i = \boldsymbol{A}$, $\forall i$ are constant matrices and a single-stage Runge–Kutta scheme is employed.

Proof. Single-stage Runge–Kutta schemes are characterized by $s = 1$ and so they satisfy the conditions of Theorem 4.3 with $a = a_{11}$. □

5. Equivalence conditions

This section performs theoretical analysis that highlights cases in which Galerkin and LSPG ROMs are equivalent, in which case the Galerkin ROM exhibits *both* continuous and discrete optimality. This suggests that Galerkin projection can be used safely in these scenarios. Section 5.1 shows that equivalence holds for explicit time integrators, Section 5.2 demonstrates equivalence in the limit of $\Delta t \to 0$, and Section 5.3 shows equivalence in the case of symmetric-positive-definite residual Jacobians.

5.1. Equivalence for explicit integrators

Corollary 5.1 (Equivalence: explicit linear multistep scheme).
Galerkin projection is equivalent to LSPG projection with $\boldsymbol{A} = \frac{1}{\sqrt{\alpha_0}} \boldsymbol{I}$ for explicit linear multistep schemes.

Proof. In the case of explicit linear multistep schemes, $\beta_0 = 0$ and so Galerkin projection corresponds to Case 3 of Theorem 4.2 with $\boldsymbol{A} = \frac{1}{\sqrt{\alpha_0}} \boldsymbol{I}$, as $\boldsymbol{\Psi} = \boldsymbol{\Phi}$ in this case. □

Corollary 5.2 (Equivalence: explicit Runge–Kutta scheme). *Galerkin projection is equivalent to LSPG projection with $\boldsymbol{A} = \boldsymbol{I}$ for explicit Runge–Kutta schemes.*

Proof. In the case of explicit Runge–Kutta schemes, $a_{ij} = 0$ $\forall j \geq i$ and so Galerkin projection corresponds to Theorem 4.3 with $\boldsymbol{A} = \boldsymbol{I}$ and $a = 0$, as $\boldsymbol{\Psi} = \boldsymbol{\Phi}$ in this case. □

5.2. Equivalence in the limit of $\Delta t \to 0$

Theorem 5.3 (Limiting equivalence).
In the limit of $\Delta t \to 0$, Galerkin projection is equivalent to LSPG projection with $\boldsymbol{A} = \frac{1}{\sqrt{\alpha_0}}\boldsymbol{I}$ for linear multistep schemes and $\boldsymbol{A}_i = I$, $i \in \mathbb{N}(s)$ for Runge–Kutta schemes.

Proof. Linear multistep schemes. Consider solving the LSPG OΔE (4.3) with $\boldsymbol{A} = \frac{1}{\sqrt{\alpha_0}}\boldsymbol{I}$. Then, the test basis defined in Eq. (4.4) is simply

$$\boldsymbol{\Psi}^n(\hat{\boldsymbol{w}}) = \frac{1}{\alpha_0}\frac{\partial \boldsymbol{r}^n}{\partial \boldsymbol{w}}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{w}}\right)\boldsymbol{\Phi}.$$

From Eq. (2.4), we can write the residual Jacobian as

$$\frac{\partial \boldsymbol{r}^n}{\partial \boldsymbol{w}}(\boldsymbol{u}) = \alpha_0\boldsymbol{I} - \Delta t\beta_0\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{\xi}}(\boldsymbol{u}, t^n).$$

Therefore, we have

$$\lim_{\Delta t\to 0}\boldsymbol{\Psi}^n(\hat{\boldsymbol{w}}) = \lim_{\Delta t\to 0}\frac{1}{\alpha_0}\left(\alpha_0\boldsymbol{I} - \Delta t\beta_0\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{\xi}}(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{w}}, t^n)\right)\boldsymbol{\Phi} = \boldsymbol{\Phi}$$

and so in the limit of $\Delta t \to 0$, the LSPG ROM solution satisfies

$$\lim_{\Delta t\to 0}\boldsymbol{\Psi}^n(\hat{\boldsymbol{w}})^T\boldsymbol{r}^n\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{w}}^n\right) = \boldsymbol{\Phi}^T\boldsymbol{r}^n\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{w}}^n\right) = 0. \tag{5.1}$$

Because the Galerkin ROM solution also satisfies Eq. (5.1) (see Eq. (3.12) of Theorem 3.4), the two techniques are equivalent in this limit, which is the desired result.
Runge–Kutta schemes. Consider solving the LSPG OΔE (4.7) with $\boldsymbol{A}_i = \boldsymbol{I}$, $i \in \mathbb{N}(s)$. Then, the test basis defined in Eq. (4.8) is simply

$$\boldsymbol{\Psi}^n_{ij}(\hat{\boldsymbol{w}}_1, \ldots, \hat{\boldsymbol{w}}_s) = \frac{\partial \boldsymbol{r}^n_i}{\partial \boldsymbol{w}_j}(\boldsymbol{\Phi}\hat{\boldsymbol{w}}_1, \ldots, \boldsymbol{\Phi}\hat{\boldsymbol{w}}_s)\boldsymbol{\Phi}.$$

Now, from Eq. (2.6) the Jacobian can be expressed as

$$\frac{\partial \boldsymbol{r}^n_i}{\partial \boldsymbol{w}_j}(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_s) = \boldsymbol{I}\delta_{ij} - \Delta t a_{ij}\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{\xi}}(\boldsymbol{x}^{n-1} + \Delta t\sum_{j=1}^{s} a_{ij}\boldsymbol{u}_j, t^{n-1} + c_i\Delta t).$$

Therefore, we have

$$\lim_{\Delta t\to 0}\boldsymbol{\Psi}^n_{ij}(\hat{\boldsymbol{w}}_1, \ldots, \hat{\boldsymbol{w}}_s) = \lim_{\Delta t\to 0}\left(\boldsymbol{I}\delta_{ij} - \Delta t a_{ij}\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{\xi}}(\boldsymbol{x}^{n-1} + \Delta t\sum_{j=1}^{s} a_{ij}\boldsymbol{u}_j, t^{n-1} + c_i\Delta t)\right)\boldsymbol{\Phi} = \boldsymbol{\Phi}\delta_{ij}$$

and so in the limit of $\Delta t \to 0$, the LSPG ROM solution satisfies

$$\lim_{\Delta t\to 0}\sum_{j=1}^{s}\boldsymbol{\Psi}^n_{ij}(\hat{\boldsymbol{w}}_1, \ldots, \hat{\boldsymbol{w}}_s)^T\boldsymbol{r}^n_j\left(\boldsymbol{\Phi}\hat{\boldsymbol{w}}^n_1, \ldots, \boldsymbol{\Phi}\hat{\boldsymbol{w}}^n_s\right) = \boldsymbol{\Phi}^T\boldsymbol{r}^n_i\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{w}}^n\right) = 0, \quad i \in \mathbb{N}(s). \tag{5.2}$$

Because the Galerkin ROM solution also satisfies Eq. (5.2) (see Eq. (3.13) of Theorem 3.4), the two techniques are equivalent in this limit, which is the desired result. □

5.3. Equivalence for symmetric-positive-definite residual Jacobians

Theorem 5.4 (Equivalence: linear multistep schemes). *In the case of linear multistep schemes, Galerkin projection is equivalent to LSPG projection with $\boldsymbol{A}(\boldsymbol{z}) = \boldsymbol{U}(\boldsymbol{z})$, where $\boldsymbol{U}$ is the Cholesky factor*[2] *of the residual-Jacobian inverse*

$$\left[\frac{\partial \boldsymbol{r}^n}{\partial \boldsymbol{w}}\right]^{-1} = \boldsymbol{U}^T\boldsymbol{U}, \tag{5.3}$$

if $\partial \boldsymbol{r}^n/\partial \boldsymbol{w}(\boldsymbol{w}^n, t^n) = \alpha_0 \boldsymbol{I} - \Delta t \beta_0 \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{\xi}}(\boldsymbol{w}^n, t^n)$ is symmetric positive definite and if

$$\frac{\partial u_{i\ell}}{\partial w_k}\phi_{kj} r_\ell^n = 0, \quad \forall i, k. \tag{5.4}$$

Here, index notation has been used.

Proof. Under the stated assumptions, the LSPG test basis defined in Eq. (4.4) is equal to the trial basis, i.e., $\boldsymbol{\Psi}^n(\hat{\boldsymbol{w}}^n) = \boldsymbol{\Phi}$. By invoking Eq. (3.12), we can see that the OΔEs for the the LSPG ROM (4.3) and Galerkin ROM (3.7) both satisfy $\boldsymbol{\Phi}^T \boldsymbol{r}^n\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{w}}^n\right) = 0$, which is the desired result. □

Theorem 5.5 (Equivalence: diagonally implicit Runge–Kutta schemes). *In the case of diagonally implicit Runge–Kutta schemes, Galerkin projection is equivalent to LSPG projection with $\boldsymbol{A}_i(\boldsymbol{z}) = \underline{\boldsymbol{U}}_i(\boldsymbol{z})$, where $\underline{\boldsymbol{U}}_i$ is the Cholesky factor of the residual-Jacobian inverse*

$$\left[\frac{\partial \boldsymbol{q}_i^n}{\partial \boldsymbol{w}}\right]^{-1} = \underline{\boldsymbol{U}}_i^T \underline{\boldsymbol{U}}_i, \tag{5.5}$$

if $\partial \boldsymbol{q}_i^n/\partial \boldsymbol{w}(\boldsymbol{w}^n, t^n) = \boldsymbol{I} - \Delta t a_{ii} \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{\xi}}\left(\boldsymbol{x}^{n-1} + \Delta t a_{ii}\boldsymbol{w}^n + \Delta t \sum_{j=1}^{i-1} a_{ij}\boldsymbol{w}_j^n, t^{n-1} + c_i \Delta t\right)$ is symmetric positive definite and if

$$\frac{\partial [\underline{\boldsymbol{U}}_i]_{j\ell}}{\partial w_m}\phi_{mk}[\boldsymbol{q}_i^n]_\ell^n = 0, \quad \forall j, k. \tag{5.6}$$

Here, index notation has been used.

Proof. Under the stated assumptions, the LSPG test basis defined in Eq. (4.13) is equal to the trial basis, i.e., $\boldsymbol{\Psi}_i^n(\hat{\boldsymbol{w}}_i^n) = \boldsymbol{\Phi}$, $i \in \mathbb{N}(s)$. By invoking Eq. (3.13), we can see that the OΔEs for the the LSPG ROM (4.12) and Galerkin ROM (3.9) both satisfy $\boldsymbol{\Phi}^T \boldsymbol{q}_i^n\left(\boldsymbol{\Phi}\hat{\boldsymbol{w}}_i^n\right) = 0$, $i = 1, \ldots, s$, which is the desired result. □

Theorem 5.6 (Equivalence: implicit Runge–Kutta schemes). *In the case of Runge–Kutta schemes, Galerkin projection exhibits discrete optimality if $\partial \bar{\boldsymbol{r}}^n/\partial \bar{\boldsymbol{w}}(\bar{\boldsymbol{w}}^n, t^n)$ is symmetric positive definite and if*

$$\frac{\partial \bar{u}_{i\ell}}{\partial \bar{w}_k}\bar{\phi}_{kj}\bar{r}_\ell^n = 0, \quad \forall i, k. \tag{5.7}$$

Here, index notation has been used and $\bar{\boldsymbol{U}} \in \mathbb{R}^{sN \times sN}$ is the Cholesky factor of the residual-Jacobian inverse, i.e.,

$$\left[\frac{\partial \bar{\boldsymbol{r}}^n}{\partial \bar{\boldsymbol{w}}}\right]^{-1} = \bar{\boldsymbol{U}}^T\bar{\boldsymbol{U}}. \tag{5.8}$$

Here,

$$\bar{\boldsymbol{w}} := \begin{bmatrix} \boldsymbol{w}_1 \\ \vdots \\ \boldsymbol{w}_s \end{bmatrix} \in \mathbb{R}^{sN}, \quad \bar{\boldsymbol{r}}^n : \bar{\boldsymbol{w}} \mapsto \begin{bmatrix} \boldsymbol{r}_1^n(\boldsymbol{w}_1, \ldots, \boldsymbol{w}_s) \\ \vdots \\ \boldsymbol{r}_s^n(\boldsymbol{w}_1, \ldots, \boldsymbol{w}_s) \end{bmatrix} \in \mathbb{R}^{sN}, \qquad \bar{\boldsymbol{\Phi}} := \begin{bmatrix} \boldsymbol{\Phi} & & \\ & \ddots & \\ & & \boldsymbol{\Phi} \end{bmatrix} \in \mathbb{R}^{sN \times sp}.$$

[2] Its derivative can be computed by solving the Lyapunov equation $\frac{\partial \boldsymbol{U}}{\partial w_k}^T \boldsymbol{U} + \boldsymbol{U}\frac{\partial \boldsymbol{U}}{\partial w_k} = -\left[\frac{\partial \boldsymbol{r}^n}{\partial \boldsymbol{w}}\right]^{-1} \frac{\partial^2 \boldsymbol{r}^n}{\partial \boldsymbol{w}\partial w_k}\left[\frac{\partial \boldsymbol{r}^n}{\partial \boldsymbol{w}}\right]^{-1}$.

Proof. First, note that solution $(\hat{\boldsymbol{w}}_1^n, \ldots, \hat{\boldsymbol{w}}_s^n)$ to the Galerkin OΔE (3.13) equivalently satisfies

$$\bar{\boldsymbol{\Phi}}^T \bar{\boldsymbol{r}}^n \left(\bar{\boldsymbol{\Phi}} \hat{\bar{\boldsymbol{w}}}^n\right) = 0,$$

where

$$\hat{\bar{\boldsymbol{w}}} := \begin{bmatrix} \hat{\boldsymbol{w}}_1 \\ \vdots \\ \hat{\boldsymbol{w}}_s \end{bmatrix} \in \mathbb{R}^{sp}.$$

We are now precisely in the situation of Theorem 5.4: the Galerkin solution is the solution to the (discrete) optimization problem

$$\underset{\boldsymbol{z} \in \operatorname{range}\left(\bar{\boldsymbol{\Phi}}\right)}{\text{minimize}} \|\bar{\boldsymbol{U}}\left(\boldsymbol{z}\right) \bar{\boldsymbol{r}}^n\left(\boldsymbol{z}\right)\|_2^2 \tag{5.9}$$

under the assumed conditions. This objective function can be written equivalently as

$$\|\bar{\boldsymbol{U}}\left(\boldsymbol{z}\right) \bar{\boldsymbol{r}}^n\left(\boldsymbol{z}\right)\|_2^2 = \sum_{i=1}^{s} \| \sum_{j=1}^{s} \bar{\boldsymbol{U}}_{ij}\left(\boldsymbol{z}_1, \ldots, \boldsymbol{z}_s\right) \boldsymbol{r}_j^n(\boldsymbol{z}_1, \ldots, \boldsymbol{z}_s)\|_2^2, \tag{5.10}$$

where $\bar{\boldsymbol{U}}_{ij} \in \mathbb{R}^{N \times N}$ denotes the (i,j) block of $\bar{\boldsymbol{U}}$. Therefore, the Galerkin ROM solution satisfies

$$(\hat{\boldsymbol{w}}_1^n, \ldots, \hat{\boldsymbol{w}}_s^n) = \arg \min_{(\hat{\boldsymbol{z}}_1, \ldots, \hat{\boldsymbol{z}}_s) \in \mathbb{R}^{p \times s}} \sum_{i=1}^{s} \| \sum_{j=1}^{s} \boldsymbol{A}_{ij}\left(\boldsymbol{\Phi}\hat{\boldsymbol{z}}_1, \ldots, \boldsymbol{\Phi}\hat{\boldsymbol{z}}_s\right) \boldsymbol{r}_j^n\left(\boldsymbol{\Phi}\hat{\boldsymbol{z}}_1, \ldots, \boldsymbol{\Phi}\hat{\boldsymbol{z}}_s\right)\|_2^2, \tag{5.11}$$

where $\boldsymbol{A}_{ij} = \bar{\boldsymbol{U}}_{ij}$. Comparing Eqs. (5.11) and (4.6) reveals that the Galerkin ROM satisfies a slightly more general notion of discrete optimality than the LSPG schemes considered in this work. □

This analysis demonstrates that Galerkin projection exhibits discrete optimality when the residual Jacobian is symmetric positive definite. This is aligned with recent work that has shown Galerkin projection to be effective for Lagrangian dynamical systems [39, 22, 23]—which are characterized by symmetric-positive-definite residual Jacobians—due to the fact that Galerkin projection preserves properties such as symplectic time evolution and energy conservation. For these reasons, using Galerkin projection is sensible for problems exhibiting these characteristics.

6. Error analysis

Ultimately, we are interested in assessing the state-space error between the (computed) time-discrete ROM solution and the (unknown) time-continuous FOM solution. This error comprises two contributions: the state-space error between (1) the time-continuous FOM and time-discrete FOM solutions (i.e., time-discretization error), and (2) the time-discrete ROM and time-discrete FOM solutions. This section focuses on the latter and performs time-discrete state-space error analyses for Galerkin and LSPG ROMs applied to different time integrators.

Section 6.1 derives error bounds for the Galerkin and LSPG ROMs for linear multistep schemes. Here, Theorem 6.1 provides *a posteriori* bounds that depend on the ROM solution, while Theorem 6.2 reports *a priori* bounds. Section 6.2 provides *a posteriori* error bounds for the Backward Euler scheme, as well as additional analyses that highlight the important role of the time step in the LSPG ROM, which is discussed in Remark 6.6. Section 6.3 derives ROM error bounds for Runge–Kutta schemes. Here, Theorem 6.7 provides *a posteriori* bounds, Corollary 6.8 specializes these results to explicit Runge–Kutta and DIRK schemes, and Theorem 6.9 and Corollary 6.10 report *a priori* bounds for Runge–Kutta schemes.

6.1. Linear multistep schemes

Here, we perform error analysis for implicit linear multistep schemes. We will use subscripts $\star$, G and P to denote the solution to full-order model OΔE (2.3), Galerkin ROM OΔE (3.7), and the LSPG ROM OΔE (4.3), respectively. We also acknowledge that linear multistep schemes with $k > 1$ usually employ different coefficients β_j and α_j for different time steps; this is necessary because at time step n, a maximum

of $n+1$ states is available from past history (starting with the initial condition at $n=0$). Therefore, we allow for coefficients that depend on the time step n, i.e., α_j^n and β_j^n. In addition, we define $\boldsymbol{\Psi}^n := \boldsymbol{\Psi}^n(\hat{\boldsymbol{x}}_P^n)$ whose entries are defined by Eq. (4.4). We can then write the discrete equations arising at each time step n for linear multistep schemes as

$$\alpha_0^n \boldsymbol{x}_\star^n = \beta_0^n \Delta t \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{x}_\star^n, t^n\right) + \boldsymbol{r}_\star\left[\boldsymbol{x}_\star^{n-k}, \ldots, \boldsymbol{x}_\star^{n-1}\right], \qquad \boldsymbol{x}_\star^0 = \boldsymbol{0} \tag{6.1}$$

$$\alpha_0^n \hat{\boldsymbol{x}}_G^n = \beta_0^n \Delta t \boldsymbol{\Phi}^T \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_G^n, t^n\right) + \hat{\boldsymbol{r}}_G\left[\hat{\boldsymbol{x}}_G^{n-k}, \ldots, \hat{\boldsymbol{x}}_G^{n-1}\right], \qquad \hat{\boldsymbol{x}}_G^0 = \boldsymbol{0} \tag{6.2}$$

$$\alpha_0^n \hat{\boldsymbol{x}}_P^n = \beta_0^n \Delta t \left((\boldsymbol{\Psi}^n)^T \boldsymbol{\Phi}\right)^{-1} (\boldsymbol{\Psi}^n)^T \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^n, t^n\right) + \hat{\boldsymbol{r}}_P^n\left[\hat{\boldsymbol{x}}_P^{n-k}, \ldots, \hat{\boldsymbol{x}}_P^{n-1}\right], \qquad \hat{\boldsymbol{x}}_P^0 = \boldsymbol{0}, \tag{6.3}$$

where

$$\begin{aligned}
\boldsymbol{r}_\star\left[\boldsymbol{x}^{n-k}, \ldots, \boldsymbol{x}^{n-1}\right] &:= \sum_{\ell=1}^{k}\left(\beta_\ell^n \Delta t \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{x}^{n-\ell}, t^{n-\ell}\right) - \alpha_\ell^n \boldsymbol{x}^{n-\ell}\right), \\
\hat{\boldsymbol{r}}_G\left[\hat{\boldsymbol{x}}^{n-k}, \ldots, \hat{\boldsymbol{x}}^{n-1}\right], &:= \sum_{\ell=1}^{k}\left(\beta_\ell^n \Delta t \boldsymbol{\Phi}^T \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}^{n-\ell}, t^{n-\ell}\right) - \alpha_\ell^n \hat{\boldsymbol{x}}^{n-\ell}\right), \\
\hat{\boldsymbol{r}}_P^n\left[\hat{\boldsymbol{x}}^{n-k}, \ldots, \hat{\boldsymbol{x}}^{n-1}\right] &:= \sum_{\ell=1}^{k}\left(\beta_\ell^n \Delta t \left((\boldsymbol{\Psi}^n)^T \boldsymbol{\Phi}\right)^{-1} (\boldsymbol{\Psi}^n)^T \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}^{n-\ell}, t^{n-\ell}\right) - \alpha_\ell^n \hat{\boldsymbol{x}}^{n-\ell}\right),
\end{aligned} \tag{6.4}$$

and $\boldsymbol{x}_\star^k := \boldsymbol{x}_\star^k - \boldsymbol{x}_0$. We define the Galerkin and LSPG operators as

$$\mathbb{V} := \boldsymbol{\Phi}\boldsymbol{\Phi}^T, \quad \text{and} \quad \mathbb{P}^n := \boldsymbol{\Phi}\left((\boldsymbol{\Psi}^n)^T \boldsymbol{\Phi}\right)^{-1} (\boldsymbol{\Psi}^n)^T,$$

respectively, and Galerkin and LSPG state-space errors at time instance n as

$$\delta \boldsymbol{x}_G^n := \boldsymbol{x}_\star^n - \boldsymbol{\Phi}\hat{\boldsymbol{x}}_G^n, \quad \text{and} \quad \delta \boldsymbol{x}_P^n := \boldsymbol{x}_\star^n - \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^n,$$

respectively. As the second argument in $\boldsymbol{f}$ does not play any role for linear multistep schemes (the time index always matches that of the first argument), will drop it for notational convenience in this section and in Section 6.2. Moreover, we assume Lipschitz continuity of $\boldsymbol{f}$ in the first argument:

$(\mathbf{A_1})$ There exist a constant $\kappa > 0$ such that for $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^N$

$$\left\|\boldsymbol{f}(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{y})\right\|_2 \leq \kappa \|\boldsymbol{x} - \boldsymbol{y}\|_2 .$$

Theorem 6.1 (*a posteriori* error bounds: linear multistep schemes). *If* $(\mathbf{A_1})$ *holds and* $\Delta t < \left|\alpha_0^j\right| / (\left|\beta_0^j\right| \kappa)$, $\forall j \in \mathbb{N}(n)$ *then*

$$\|\delta \boldsymbol{x}_G^n\|_2 \leq \sum_{j=0}^{n-1} \sum_{\ell=0}^{\min(k,j)} \left[\mathbf{1}_{\{0\}}(j-\ell) + \sum_{(\eta_i)\in\mathcal{A}(j-\ell)} \prod_{i=1}^{|(\eta_i)|} \gamma_{\eta_i}^{n-\sum_{m=1}^{i-1}\eta_m}\right] \varepsilon_\ell^{n-j+\ell} \left\|(\boldsymbol{I} - \mathbb{V})\, \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_G^{n-j}\right)\right\|_2 \tag{6.5}$$

$$\|\delta \boldsymbol{x}_P^n\|_2 \leq \sum_{j=0}^{n-1} \sum_{\ell=0}^{\min(k,j)} \left[\mathbf{1}_{\{0\}}(j-\ell) + \sum_{(\eta_i)\in\mathcal{A}(j-\ell)} \prod_{i=1}^{|(\eta_i)|} \gamma_{\eta_i}^{n-\sum_{m=1}^{i-1}\eta_m}\right] \varepsilon_\ell^{n-j+\ell} \left\|\left(\boldsymbol{I} - \mathbb{P}^{n-j+\ell}\right) \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{n-j}\right)\right\|_2, \tag{6.6}$$

where we have defined $\varepsilon_\ell^k := \left|\beta_\ell^k\right| \Delta t / h^k$, $\gamma_\ell^k := (\left|\beta_\ell^k\right| \kappa \Delta t + \left|\alpha_\ell^n\right|)/h^k$, *and* $h^k := \left|\alpha_0^k\right| - \left|\beta_0^k\right| \kappa \Delta t$. *Here,* $\mathbf{1}_A(x)$ *denotes the indicator function,* $\mathcal{A}(p) := \{(\eta_i) \mid \eta_i \in \mathbb{N}(k),\ \sum_i \eta_i = p\}$, *and* $|(\eta_i)|$ *denotes the length of the tuple* (η_i).

Proof. It is enough to show bound (6.6), as the arguments for (6.5) are similar. Let n be fixed but arbitrary, then subtracting Eq. (6.3) from Eq. (6.1) yields

$$|\alpha_0^n| \|\delta \boldsymbol{x}_P^n\|_2 \leq |\beta_0^n| \, \Delta t \left\| \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{x}_\star^n\right) - \mathbb{P}^n \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^n\right)\right\|_2 + \left\|\delta \boldsymbol{r}_P^{n-1}\right\|_2, \tag{6.7}$$

where $\delta \boldsymbol{r}_P^{n-1} := \boldsymbol{r}_\star\left[\boldsymbol{x}_\star^{n-k}, \ldots, \boldsymbol{x}_\star^{n-1}\right] - \boldsymbol{\Phi}\hat{\boldsymbol{r}}_P^n\left[\hat{\boldsymbol{x}}_P^{n-k}, \ldots, \hat{\boldsymbol{x}}_P^{n-1}\right]$. Adding and subtracting $\boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^n\right)$ and applying the triangle inequality leads to

$$|\alpha_0^n| \|\delta \boldsymbol{x}_P^n\|_2 \leq |\beta_0^n| \, \Delta t \left(\left\|\left(\boldsymbol{I} - \mathbb{P}^n\right) \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^n\right)\right\|_2 + \left\| \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{x}_\star^n\right) - \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^n\right)\right\|_2 \right) + \left\|\delta \boldsymbol{r}_P^{n-1}\right\|_2. \tag{6.8}$$

Invoking $(\mathbf{A_1})$, and using $\Delta t < |\alpha_0^n| / |\beta_0^n| \, \kappa$, we deduce

$$\|\delta \boldsymbol{x}_P^n\|_2 \leq \frac{|\beta_0^n| \, \Delta t}{h^n} \left\|\left(\boldsymbol{I} - \mathbb{P}^n\right) \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^n\right)\right\|_2 + \frac{1}{h^n}\left\|\delta \boldsymbol{r}_P^{n-1}\right\|_2. \tag{6.9}$$

Next, we will estimate $\left\|\delta \boldsymbol{r}_P^{n-1}\right\|_2$. Using the definition of $\boldsymbol{r}_\star$, $\hat{\boldsymbol{r}}_P^n$ from (6.4) we derive

$$\left\|\delta \boldsymbol{r}_P^{n-1}\right\|_2 \leq \sum_{\ell=1}^{k} \left(|\beta_\ell^n| \, \Delta t \left\| \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{x}_\star^{n-\ell}\right) - \mathbb{P}^n \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{n-\ell}\right)\right\|_2 + |\alpha_\ell^n| \left\|\delta \boldsymbol{x}_P^{n-\ell}\right\|_2 \right). \tag{6.10}$$

Adding and subtracting $\boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{n-\ell}\right)$, applying the triangle inequality in conjunction with $(\mathbf{A_1})$ yields

$$\left\|\delta \boldsymbol{r}_P^{n-1}\right\|_2 \leq \sum_{\ell=1}^{k} |\beta_\ell^n| \, \Delta t \left\|\left(\boldsymbol{I} - \mathbb{P}^n\right) \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{n-\ell}\right)\right\|_2 + \sum_{\ell=1}^{k} \left(|\beta_\ell^n| \, \kappa \Delta t + |\alpha_\ell^n|\right) \left\|\delta \boldsymbol{x}_P^{n-\ell}\right\|_2. \tag{6.11}$$

Then (6.9) and (6.11) imply

$$\|\delta \boldsymbol{x}_P^n\|_2 \leq \sum_{\ell=0}^{k} \varepsilon_\ell^n \left\|\left(\boldsymbol{I} - \mathbb{P}^n\right) \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{n-\ell}\right)\right\|_2 + \sum_{\ell=1}^{k} \gamma_\ell^n \left\|\delta \boldsymbol{x}_P^{n-\ell}\right\|_2, \tag{6.12}$$

where we have used the definitions for ε_ℓ^n and γ_ℓ^n. We note that the corresponding (time-local) error bound for the Galerkin ROM is simply

$$\|\delta \boldsymbol{x}_G^n\|_2 \leq \sum_{\ell=0}^{k} \varepsilon_\ell^n \left\|\left(\boldsymbol{I} - \mathbb{V}\right) \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_G^{n-\ell}\right)\right\|_2 + \sum_{\ell=1}^{k} \gamma_\ell^n \left\|\delta \boldsymbol{x}_G^{n-\ell}\right\|_2, \tag{6.13}$$

Notice that the term $\left\|\left(\boldsymbol{I} - \mathbb{P}^i\right) \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{i-j}\right)\right\|_2$ in inequality (6.12) corresponds to the error introduced at time step $i \in \mathbb{N}(n)$ from the state at time step $i - j$ with $j \in \mathbb{N}(k)$; this term always appears in the time-local error bound with coefficient ε_j^i. Further, it contributes to the error at a given time step $n > i$ through appropriate products of γ_ℓ^m. For example, the product $\gamma_1^n \gamma_2^{n-1} \gamma_1^{n-3}$ provides one possible path for 'traversing' the time-local error bounds from time step n to an earlier error contribution at time step $n - 4$. Applying this notion more generally and using $\delta \boldsymbol{x}_P^0 = \mathbf{0}$, the error can be bounded by induction according to inequality (6.6). □

The bounds in Theorem 6.1 can be considered *a posteriori* error bounds, as they depend on the ROM solutions $\hat{\boldsymbol{x}}_G$ and $\hat{\boldsymbol{x}}_P$ and can thus be computed *a posteriori* if κ can be estimated. Note that the rightmost term in the Galerkin bound corresponds to the orthogonal projection error of $\boldsymbol{f}$ onto range$(\boldsymbol{\Phi})$, while the LSPG bound entails an oblique projector that depends on the ROM solution. Because this oblique projection associates with a discrete residual-minimization property, the LSPG ROM can yield smaller error bounds as will be shown in Corollary 6.4 of Section 6.2. Also, the first term within square brackets corresponds to errors incurred at the current time step n (i.e., via the leftmost term on the right-hand side of inequality (6.12)), while the second term corresponds to all possible 'paths' from current time step n to the error contribution at previous time steps (i.e., the rightmost term on the right-hand side of inequality (6.12)). We now derive *a priori* error bounds by slightly modifying the steps in the above proof.

Theorem 6.2 (*a priori* error bounds: linear multistep schemes). *If* $(\mathbf{A_1})$ *holds,* $\Delta t < \left|\alpha_0^j\right| / (\left|\beta_0^j\right| \kappa)$, $\forall j \in \mathbb{N}(n)$ *for the Galerkin ROM, and* $\Delta t < \left|\alpha_0^j\right| / (\left|\beta_0^j\right| \kappa \|\mathbb{P}^n\|)$, $\forall j \in \mathbb{N}(n)$ *for the LSPG ROM, then*

$$\|\delta \boldsymbol{x}_G^n\|_2 \leq \sum_{j=0}^{n-1} \sum_{\ell=0}^{\min(k,j)} \left[\mathbf{1}_{\{0\}}(j-\ell) + \sum_{(\eta_i)\in\mathcal{A}(j-\ell)} \prod_{i=1}^{|(\eta_i)|} \gamma_{\eta_i}^{n-\sum_{m=1}^{i-1}\eta_m} \right] \varepsilon_\ell^{n-j+\ell} \left\| (\boldsymbol{I}-\mathbb{V})\, \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{x}_\star^{n-j}\right) \right\|_2 \tag{6.14}$$

$$\|\delta \boldsymbol{x}_P^n\|_2 \leq \sum_{j=0}^{n-1} \sum_{\ell=0}^{\min(k,j)} \left[\mathbf{1}_{\{0\}}(j-\ell) + \sum_{(\eta_i)\in\mathcal{A}(j-\ell)} \prod_{i=1}^{|(\eta_i)|} \bar{\gamma}_{\eta_i}^{n-\sum_{m=1}^{i-1}\eta_m} \right] \bar{\varepsilon}_\ell^{n-j+\ell} \left\| \left(\boldsymbol{I}-\mathbb{P}^{n-j+\ell}\right) \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{x}_\star^{n-j}\right) \right\|_2, \tag{6.15}$$

where we have defined $\bar{\varepsilon}_\ell^k := \left|\beta_\ell^k\right| \Delta t / \bar{h}^k$ $\bar{\gamma}_\ell^k := (\left|\beta_\ell^k\right| \kappa \Delta t \|\mathbb{P}^k\|_2 + \left|\alpha_\ell^n\right|)/\bar{h}^k$, $\bar{h}^k := \left|\alpha_0^k\right| - \left|\beta_0^k\right| \kappa \Delta t \|\mathbb{P}^k\|_2$. *Other quantities are defined in Theorem 6.1.*

Proof. Instead of adding and subtracting $\boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^n\right)$ between Eqs. (6.7) and (6.8) and between Eqs. (6.10) and (6.11) in the proof of Theorem 6.1, adding and subtracting $\mathbb{P}^n \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{x}_\star^n\right)$ and using $\|\mathbb{V}\|_2 = 1$ yields the stated result. □

We now make two observations about Theorems 6.1 and 6.2. First, note that bound (6.15) is not quite an *a priori* bound, as the operator $\mathbb{P}^k$ depends on the LSPG ROM solution $\hat{\boldsymbol{x}}_P^k$; while this dependence could be removed, the bound in its current form facilitates comparison with the Galerkin bound. Second, note that the rightmost term in the Galerkin *a priori* error bound (6.14) will always be smaller than the rightmost term in the LSPG *a priori* error bound (6.15), as $\left\|(\boldsymbol{I}-\mathbb{V})\,\boldsymbol{f}\right\|_2$ is orthogonal projection error of a (fixed) vector $\boldsymbol{f}$ onto range$(\boldsymbol{\Phi})$. As previously discussed, this is in contrast to the *a posteriori* bounds, where the discrete residual-minimization property of the LSPG ROM can yield lower error bounds; this will be illustrated in Corollary 6.4 of Section 6.2 below.

6.2. Backward Euler

We now specialize the result in Theorem 6.1 to the case of the Backward Euler scheme.

Corollary 6.3 (*a posteriori* error bounds: Backward Euler). *Under the assumptions of Theorem 6.1, for Backward Euler we obtain*

$$\|\delta \boldsymbol{x}_G^n\|_2 \leq \Delta t \sum_{j=0}^{n-1} \frac{1}{(h)^{j+1}} \left\| (\boldsymbol{I}-\mathbb{V})\, \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_G^{n-j}\right) \right\|_2 \tag{6.16}$$

$$\|\delta \boldsymbol{x}_P^n\|_2 \leq \Delta t \sum_{j=0}^{n-1} \frac{1}{(h)^{j+1}} \left\| \left(\boldsymbol{I}-\mathbb{P}^{n-j}\right) \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{n-j}\right) \right\|_2, \tag{6.17}$$

where $h := 1 - \kappa\Delta t$.

Proof. Backward Euler is a single-step method that can be characterized by Eq. (2.2) with $k = 1$, $\alpha_0 = 1$, $\alpha_1 = -1$, $\beta_0 = 1$, and $\beta_1 = 0$. Because it is a single-step method, these coefficients do not vary between time steps; as a result, the constants appearing in Theorem 6.1 are also time-step independent and are $h = 1 - \kappa\Delta t$, $\varepsilon_0 = \Delta t/h$, $\varepsilon_1 = 0$, $\gamma_0 = (\kappa\Delta t + 1)/h$, and $\gamma_1 = 1/h$. Substituting these values into error bound (6.6) and noting that

$$\mathcal{A}(j) = \begin{cases} \{(1,\ldots,1) \in \mathbb{R}^j\} & j > 0 \\ \emptyset & \text{otherwise,} \end{cases}$$

yields

$$\|\delta \boldsymbol{x}_P^n\|_2 \leq \frac{\Delta t}{h} \sum_{j=0}^{n-1} \underbrace{\left[\mathbf{1}_{\{0\}}(j) + (1/h)^j \mathbf{1}_{\mathbb{N}(n-1)}(j)\right]}_{(1/h)^j} \left\| \left(\boldsymbol{I}-\mathbb{P}^{n-j}\right) \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{n-j}\right) \right\|_2, \tag{6.18}$$

which simplifies to bound (6.17). Derivation of bound (6.16) is identical and is thus omitted. □

It is not clear how to directly compare the Galerkin and the LSPG *a posteriori* error bounds (6.5) and (6.6), although it should be noted that both bounds grow exponentially in time due to the amplification factor. However, for the numerical experiments in Section 7, which use the three-point backward-difference scheme, the LSPG uniformly outperforms the Galerkin ROM. We now provide further theoretical justification for these numerical observations for the backward Euler scheme by comparing *a posteriori* bounds (6.16) and (6.17). Similar arguments can be applied to the more general schemes.

To this end, for $j = 0, \ldots, n-1$, it is sufficient to compare

$$\Delta t \left\| (\boldsymbol{I} - \mathbb{V})\, \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_G^{n-j}\right)\right\|_2 \quad \text{and} \quad \Delta t \left\| (\boldsymbol{I} - \mathbb{P}^n)\, \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{n-j}\right)\right\|_2 .$$

Invoking Eq. (6.2), we can rewrite the first term as

$$\Delta t \left\| (\boldsymbol{I} - \mathbb{V})\, \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_G^{n-j}\right)\right\|_2 = \left\| \boldsymbol{\Phi}\hat{\boldsymbol{x}}_G^{n-j} - \Delta t \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_G^{n-j-1}\right) - \boldsymbol{\Phi}\hat{\boldsymbol{x}}_G^{n-j-1}\right\|_2 . \tag{6.19}$$

Similarly, using Eq. (6.3) and the (discrete residual-minimizing) optimality property of $\hat{\boldsymbol{x}}_P^{n-j}$, we deduce

$$\begin{aligned} \Delta t \left\| \left(\boldsymbol{I} - \mathbb{P}^{n-j}\right) \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{n-j}\right)\right\|_2 &= \left\| \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{n-j} - \Delta t \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{n-j}\right) - \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{n-j-1}\right\|_2 \\ &= \min_{\boldsymbol{y}} \left\| \boldsymbol{\Phi}\boldsymbol{y} - \Delta t \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\boldsymbol{y}\right) - \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{n-j-1}\right\|_2 . \end{aligned} \tag{6.20}$$

This produces an interesting result: for the same previous state $\hat{\boldsymbol{x}}_P^{n-j-1} = \hat{\boldsymbol{x}}_G^{n-j-1}$, a direct comparison of Eqs. (6.19) and (6.20) yields that the quantity on the right-hand side of (6.20) will always be less than that of (6.19). We state this result below.

Corollary 6.4. *If $\hat{\boldsymbol{x}}_P^j = \hat{\boldsymbol{x}}_G^j$, $j \in \mathbb{N}(n-1)$, then under the assumptions of Theorem 6.1, the a posteriori upper bound for $\|\delta\hat{\boldsymbol{x}}_P^k\|_2$ in Eq.* (6.17) *will be less than the upper bound for $\|\delta\hat{\boldsymbol{x}}_G^k\|_2$ in Eq.* (6.16) *for $k \in \mathbb{N}(n)$.*

Corollary 6.5. *If $\bar{\boldsymbol{x}}$ solves an auxiliary problem that computes the full-space solution increment centered on the LSPG ROM trajectory*

$$\bar{\boldsymbol{x}}^j = \Delta t \boldsymbol{f}\left(\boldsymbol{x}_0 + \bar{\boldsymbol{x}}^j\right) + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{j-1}, \quad j \in \mathbb{N}(n), \tag{6.21}$$

then the following holds:

$$\|\delta\boldsymbol{x}_P^n\|_2 \leq (1 + \kappa\Delta t) \sum_{j=0}^{n-1} \frac{\mu^{n-j}}{(h)^{j+1}} \tag{6.22}$$

$$= \Delta t(1 + \kappa\Delta t) \sum_{j=0}^{n-1} \frac{\bar{\mu}^{n-j}}{(h)^{j+1}} \|\boldsymbol{f}(\bar{\boldsymbol{x}}^{n-j})\|_2. \tag{6.23}$$

Here, $\mu^j := \left\| \boldsymbol{\Phi}\Delta\hat{\boldsymbol{x}}_P^j - \Delta\bar{\boldsymbol{x}}^j \right\|_2$ denotes the difference in solution increments at time instance j, where $\Delta\hat{\boldsymbol{x}}_P^j := \hat{\boldsymbol{x}}_P^j - \hat{\boldsymbol{x}}_P^{j-1}$ and $\Delta\bar{\boldsymbol{x}}^j := \bar{\boldsymbol{x}}^j - \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{j-1}$. We denote the relative solution increment at time instance j by $\bar{\mu}^j := \mu^j / \|\Delta\bar{\boldsymbol{x}}^j\|_2$.

Proof. Eq. (6.17) in conjunction with (6.20) implies

$$\|\delta\boldsymbol{x}_P^n\|_2 \leq \sum_{j=0}^{n-1} \frac{1}{(h)^{j+1}} \left\| \boldsymbol{\Phi}\Delta\hat{\boldsymbol{x}}_P^{n-j} - \Delta t \boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{n-j}\right)\right\|_2 . \tag{6.24}$$

We can also write the auxiliary equation (6.21) as $\Delta\bar{\boldsymbol{x}}^j = \Delta t \boldsymbol{f}\left(\bar{\boldsymbol{x}}^j\right)$, $j \in \mathbb{N}(n)$, which allows us to rewrite bound (6.24) as

$$\|\delta \boldsymbol{x}_P^n\|_2 \leq \sum_{j=0}^{n-1} \frac{1}{(h)^{j+1}} \cdot \Big\| \left(\boldsymbol{\Phi}\Delta\hat{\boldsymbol{x}}_P^{n-j} - \Delta\bar{\boldsymbol{x}}^{n-j}\right) - \Delta t \left(\boldsymbol{f}\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\Delta\hat{\boldsymbol{x}}_P^{n-j} + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{n-j-1}\right) - \boldsymbol{f}\left(\boldsymbol{x}_0 + \Delta\bar{\boldsymbol{x}}^{n-j} + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{n-j-1}\right)\right)\Big\|_2.$$

Lipschitz continuity of $\boldsymbol{f}$ leads to the bound (6.22). To obtain Eq. (6.23), we multiply and divide by $\|\Delta\bar{\boldsymbol{x}}^{n-j}\|_2$ for each term in the summation and use $\Delta\bar{\boldsymbol{x}}^{n-j} = \Delta t \boldsymbol{f}\left(\bar{\boldsymbol{x}}^{n-j}\right)$. □

Corollary 6.5 is useful in that it expresses the LSPG ROM error in terms of the (time-local) single-step errors incurred by projection along the LSPG ROM trajectory. In addition, this result highlights the critical role of the time step Δt in the performance of the LSPG ROM; the following remark provides this discussion.

Remark 6.6. The time step Δt in the error bound (6.23) for the LSPG ROM solution plays an important role. In particular, decreasing the time step produces both beneficial effects (bound decrease) and deleterious effects (bound increase), which we denote by '+' and '-', respectively as follows:

+ The time-discretization error decreases (this does not appear in the time-discrete error analysis above).
- The number of overall time steps n increases, so there are more terms in the summation.
+ The terms $\Delta t(1 + \kappa\Delta t)$ and $1/(h)^{j+1}$ decrease.
? The term $\bar{\mu}^{n-j}$ may increase or decrease, depending on the spectral content of the basis $\boldsymbol{\Phi}$.

We now discuss this final ambiguous effect. The term $\bar{\mu}^n$ can be interpreted as the *relative error in solution increment over* $\left[(n-1)\Delta t, n\Delta t\right]$. Clearly, the ability of the LSPG ROM to make $\bar{\mu}^n$ small depends on the spectral content of the basis $\boldsymbol{\Phi}$: if the basis only captures modes that evolve over long time scales, then $\bar{\mu}^n$ will be large (i.e., close to one), as the basis does not contain the 'fast evolving' solution components that change over a single time step. This suggests that the time step should be 'matched' to the spectral content of the reduced basis $\boldsymbol{\Phi}$. In Section 7.5 of the experiments, we explore this issue numerically, and demonstrate that the error bound is minimized for an intermediate value of the time step Δt.

We note that the above arguments do not hold for the Galerkin ROM, which is simply an ODE that does not depend on the time step. Instead, decreasing the time step should increase accuracy, as it has the effect of reducing the time-discretization error.

6.3. Runge–Kutta schemes

We now derive Runge–Kutta error bounds for the Galerkin ROM (3.10) and the LSPG ROM (4.7). For notational simplicity, we define $\boldsymbol{f}_i^n(\cdot) := \boldsymbol{f}(\cdot, t^{n-1} + c_i\Delta t)$, $i \in \mathbb{N}(s)$, $n \in \mathbb{N}(T/\Delta t)$.

We rewrite Eqs. (2.5), (3.9), and (4.7) as

$$\boldsymbol{w}_{\star,i}^n = \boldsymbol{f}_i^n\Big(\boldsymbol{x}_0 + \boldsymbol{x}_\star^{n-1} + \Delta t \sum_{j=1}^{s} a_{ij}\boldsymbol{w}_{\star,j}^n\Big), \quad i \in \mathbb{N}(s) \qquad \boldsymbol{x}_\star^0 = \boldsymbol{0} \tag{6.25}$$

$$\hat{\boldsymbol{w}}_{G,i}^n = \boldsymbol{\Phi}^T\boldsymbol{f}_i^n\Big(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_G^{n-1} + \Delta t \sum_{j=1}^{s} a_{ij}\boldsymbol{\Phi}\hat{\boldsymbol{w}}_{G,j}^n\Big), \quad i \in \mathbb{N}(s) \qquad \boldsymbol{x}_G^0 = \boldsymbol{0} \tag{6.26}$$

$$\begin{aligned}\hat{\boldsymbol{w}}_{P,i}^n = &\left((\boldsymbol{\Psi}_{ii}^n)^T\boldsymbol{\Phi}\right)^{-1}(\boldsymbol{\Psi}_{ii}^n)^T\boldsymbol{f}_i^n\Big(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{n-1} + \Delta t \sum_{j=1}^{s} a_{ij}\boldsymbol{\Phi}\hat{\boldsymbol{w}}_{P,j}^n\Big) \\ &- \left((\boldsymbol{\Psi}_{ii}^n)^T\boldsymbol{\Phi}\right)^{-1}\sum_{e=1,e\neq i}^{s}(\boldsymbol{\Psi}_{ie}^n)^T\Big(\boldsymbol{\Phi}\hat{\boldsymbol{w}}_{P,e}^n - \boldsymbol{f}_e^n\Big(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{n-1} + \Delta t \sum_{j=1}^{s} a_{ej}\boldsymbol{\Phi}\hat{\boldsymbol{w}}_{P,j}^n\Big)\Big), \quad i \in \mathbb{N}(s) \quad \boldsymbol{x}_P^0 = \boldsymbol{0}.\end{aligned} \tag{6.27}$$

Theorem 6.7 (*a posteriori* error bounds: Runge–Kutta schemes). *If* $(\mathbf{A_1})$ *holds and* Δt *is such that*

(a) *the matrix* $\boldsymbol{D} \in \mathbb{R}^{s\times s}$ *with entries* $d_{ij} := \delta_{ij} - \kappa\Delta t|a_{ij}|$ *is invertible, and*

(b) *for every* $\boldsymbol{x}, \boldsymbol{y} \geq 0$, *if* $\boldsymbol{D}\boldsymbol{x} \leq \boldsymbol{y}$ *then* $\boldsymbol{x} \leq \boldsymbol{D}^{-1}\boldsymbol{y}$,

then

$$\begin{aligned}\|\delta\boldsymbol{x}_G^n\|_2 \leq \Delta t \sum_{\ell=0}^{n-1}\Big(1+\kappa\Delta t\sum_{k=1}^{s}|b_k|\sum_{i=1}^{s}[\boldsymbol{D}^{-1}]_{ki}\Big)^{\ell}\cdot \\ \Big(\sum_{k=1}^{s}|b_k|\sum_{i=1}^{s}[\boldsymbol{D}^{-1}]_{ki}\Big\|(\boldsymbol{I}-\mathbb{V})\boldsymbol{f}_i^{n-\ell}\Big(\boldsymbol{x}_0+\boldsymbol{\Phi}\hat{\boldsymbol{x}}_G^{n-\ell-1}+\Delta t\sum_{j=1}^{s}a_{ij}\boldsymbol{\Phi}\hat{\boldsymbol{w}}_{G,j}^{n-\ell}\Big)\Big\|_2\Big).\end{aligned} \tag{6.28}$$

and

$$\begin{aligned}\|\delta\boldsymbol{x}_P^n\|_2 \leq \Delta t \sum_{\ell=0}^{n-1}\Big(1+\kappa\Delta t\sum_{k=1}^{s}|b_k|\sum_{i=1}^{s}[\boldsymbol{D}^{-1}]_{ki}\Big)^{\ell}\cdot \\ \Bigg(\sum_{k=1}^{s}|b_k|\sum_{i=1}^{s}[\boldsymbol{D}^{-1}]_{ki}\Big\|(\boldsymbol{I}-\mathbb{P}_i^{n-\ell})\boldsymbol{f}_i^{n-\ell}\Big(\boldsymbol{x}_0+\boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{n-\ell-1}+\Delta t\sum_{j=1}^{s}a_{ij}\boldsymbol{\Phi}\hat{\boldsymbol{w}}_{P,j}^{n-\ell}\Big)\Big\|_2 \\ +\sum_{k=1}^{s}|b_k|\sum_{i=1}^{s}[\boldsymbol{D}^{-1}]_{ki}\Big\|\boldsymbol{\Phi}\left((\boldsymbol{\Psi}_{ii}^{n-\ell})^T\boldsymbol{\Phi}\right)^{-1}\sum_{e=1,e\neq i}^{s}(\boldsymbol{\Psi}_{ie}^{n-\ell})^T\Big(\boldsymbol{\Phi}\hat{\boldsymbol{w}}_{P,e}^{n-\ell}-\boldsymbol{f}_e^{n-\ell}\Big(\boldsymbol{x}_0+\boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{n-\ell-1}+\Delta t\sum_{j=1}^{s}a_{ej}\boldsymbol{\Phi}\hat{\boldsymbol{w}}_{P,j}^{n-\ell}\Big)\Big)\Big\|_2\Bigg),\end{aligned} \tag{6.29}$$

where $\mathbb{P}_i^n := \boldsymbol{\Phi}\left((\boldsymbol{\Psi}_{ii}^n)^T\boldsymbol{\Phi}\right)^{-1}(\boldsymbol{\Psi}_{ii}^n)^T$. *Here, inequalities applied to vectors hold entrywise.*

Proof. Galerkin ROM. First we will show bound (6.28). Subtracting Eq. (6.26) from Eq. (6.25) and applying the triangle inequality yields

$$\|\delta\boldsymbol{w}_{G,i}^n\|_2 \leq \Big\|\boldsymbol{f}_i^n\Big(\boldsymbol{x}_0+\boldsymbol{x}_\star^{n-1}+\Delta t\sum_{j=1}^{s}a_{ij}\boldsymbol{w}_{\star,j}^n\Big)-\mathbb{V}\boldsymbol{f}_i^n\Big(\boldsymbol{x}_0+\boldsymbol{\Phi}\hat{\boldsymbol{x}}_G^{n-1}+\Delta t\sum_{j=1}^{s}a_{ij}\boldsymbol{\Phi}\hat{\boldsymbol{w}}_{G,j}^n\Big)\Big\|_2, \qquad i\in\mathbb{N}(s),$$

where $\delta\boldsymbol{w}_{G,i}^n := \boldsymbol{w}_{\star,i}^n - \boldsymbol{\Phi}\hat{\boldsymbol{w}}_{G,i}^n$. Adding and subtracting $\boldsymbol{f}_i^n\Big(\boldsymbol{x}_0+\boldsymbol{\Phi}\hat{\boldsymbol{x}}_G^{n-1}+\Delta t\sum_{j=1}^{s}a_{ij}\boldsymbol{\Phi}\hat{\boldsymbol{w}}_{G,j}^n\Big)$ and invoking assumption $(\mathbf{A_1})$, we deduce

$$\|\delta\boldsymbol{w}_{G,i}^n\|_2-\kappa\Delta t\sum_{j=1}^{s}|a_{ij}|\|\delta\boldsymbol{w}_{G,j}^n\|_2 \leq \Big\|(\boldsymbol{I}-\mathbb{V})\boldsymbol{f}_i^n\Big(\boldsymbol{x}_0+\boldsymbol{\Phi}\hat{\boldsymbol{x}}_G^{n-1}+\Delta t\sum_{j=1}^{s}a_{ij}\boldsymbol{\Phi}\hat{\boldsymbol{w}}_{G,j}^n\Big)\Big\|_2+\kappa\|\delta\boldsymbol{x}_G^{n-1}\|_2, \qquad i\in\mathbb{N}(s).$$

Selecting Δt small enough such that (a) and (b) hold yields

$$\eta := \|\delta\boldsymbol{w}_{G,k}^n\|_2 \leq \sum_{i=1}^{s}[\boldsymbol{D}^{-1}]_{ki}\Big\|(\boldsymbol{I}-\mathbb{V})\boldsymbol{f}_i^n\Big(\boldsymbol{x}_0+\boldsymbol{\Phi}\hat{\boldsymbol{x}}_G^{n-1}+\Delta t\sum_{j=1}^{s}a_{ij}\boldsymbol{\Phi}\hat{\boldsymbol{w}}_{G,j}^n\Big)\Big\|_2+\kappa\|\delta\boldsymbol{x}_G^{n-1}\|_2\sum_{i=1}^{s}[\boldsymbol{D}^{-1}]_{ki},$$

where $[\cdot]_{ij}$ denotes entry (i,j) of the argument. From explicit state updates (2.7) and (3.11), we obtain

$$\|\delta\boldsymbol{x}_G^n\|_2 \leq \|\delta\boldsymbol{x}_G^{n-1}\|_2+\Delta t\sum_{k=1}^{s}|b_k|\|\delta\boldsymbol{w}_{G,k}^n\|_2.$$

Using the upper bound for η yields

$$\begin{aligned}\|\delta\boldsymbol{x}_G^n\|_2 \leq &\Big(1+\kappa\Delta t\sum_{k=1}^{s}|b_k|\sum_{i=1}^{s}[\boldsymbol{D}^{-1}]_{ki}\Big)\|\delta\boldsymbol{x}_G^{n-1}\|_2 \\ &+\Delta t\sum_{k=1}^{s}|b_k|\sum_{i=1}^{s}[\boldsymbol{D}^{-1}]_{ki}\Big\|(\boldsymbol{I}-\mathbb{V})\boldsymbol{f}_i^n\Big(\boldsymbol{x}_0+\boldsymbol{\Phi}\hat{\boldsymbol{x}}_G^{n-1}+\Delta t\sum_{j=1}^{s}a_{ij}\boldsymbol{\Phi}\hat{\boldsymbol{w}}_{G,j}^n\Big)\Big\|_2.\end{aligned}$$

Finally, an induction argument produces the desired result (6.28).
LSPG ROM. We now prove bound (6.29). Subtracting (6.27) from (6.25) and applying the triangle inequality yields

$$\|\delta \boldsymbol{w}_{P,i}^n\|_2 \leq \left\| \boldsymbol{f}_i^n\left(\boldsymbol{x}_0 + \boldsymbol{x}_\star^{n-1} + \Delta t \sum_{j=1}^s a_{ij} \boldsymbol{w}_{\star,j}^n\right) - \mathbb{P}_i^n \boldsymbol{f}_i^n\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{n-1} + \Delta t \sum_{j=1}^s a_{ij} \boldsymbol{\Phi}\hat{\boldsymbol{w}}_{P,j}^n\right)\right\|_2$$
$$+ \left\| \boldsymbol{\Phi}\left((\boldsymbol{\Psi}_{ii}^n)^T \boldsymbol{\Phi}\right)^{-1} \sum_{e=1, e\neq i}^s (\boldsymbol{\Psi}_{ie}^n)^T \left(\boldsymbol{\Phi}\hat{\boldsymbol{w}}_{P,e}^n - \boldsymbol{f}_e^n\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{n-1} + \Delta t \sum_{j=1}^s a_{ej} \boldsymbol{\Phi}\hat{\boldsymbol{w}}_{P,j}^n\right)\right)\right\|_2, \qquad i \in \mathbb{N}(s).$$

Adding and subtracting $\boldsymbol{f}_i^n\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{n-1} + \Delta t \sum_{j=1}^s a_{ij} \boldsymbol{\Phi}\hat{\boldsymbol{w}}_{P,j}^n\right)$ and invoking assumption $(\mathbf{A_1})$, for $i \in \mathbb{N}(s)$ we deduce

$$\|\delta \boldsymbol{w}_{P,i}^n\|_2 - \kappa \Delta t \sum_{j=1}^s |a_{ij}| \|\delta \boldsymbol{w}_{P,j}^n\|_2 \leq \left\|(\boldsymbol{I} - \mathbb{P}_i^n)\boldsymbol{f}_i^n\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{n-1} + \Delta t \sum_{j=1}^s a_{ij} \boldsymbol{\Phi}\hat{\boldsymbol{w}}_{P,j}^n\right)\right\|_2 + \kappa \|\delta \boldsymbol{x}_P^{n-1}\|_2$$
$$+ \left\| \boldsymbol{\Phi}\left((\boldsymbol{\Psi}_{ii}^n)^T \boldsymbol{\Phi}\right)^{-1} \sum_{e=1, e\neq i}^s (\boldsymbol{\Psi}_{ie}^n)^T \left(\boldsymbol{\Phi}\hat{\boldsymbol{w}}_{P,e}^n - \boldsymbol{f}_e^n\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{n-1} + \Delta t \sum_{j=1}^s a_{ej} \boldsymbol{\Phi}\hat{\boldsymbol{w}}_{P,j}^n\right)\right)\right\|_2.$$

Again selecting Δt small enough such that (a) and (b) hold yields

$$\zeta := \|\delta \boldsymbol{w}_{P,k}^n\|_2 \leq \sum_{i=1}^s [\boldsymbol{D}^{-1}]_{ki} \left\|(\boldsymbol{I} - \mathbb{P}_i^n)\boldsymbol{f}_i^n\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{n-1} + \Delta t \sum_{j=1}^s a_{ij} \boldsymbol{\Phi}\hat{\boldsymbol{w}}_{P,j}^n\right)\right\|_2 + \kappa \|\delta \boldsymbol{x}_P^{n-1}\|_2 \sum_{i=1}^s [\boldsymbol{D}^{-1}]_{ki}$$
$$+ \sum_{i=1}^s [\boldsymbol{D}^{-1}]_{ki} \left\| \boldsymbol{\Phi}\left((\boldsymbol{\Psi}_{ii}^n)^T \boldsymbol{\Phi}\right)^{-1} \sum_{e=1, e\neq i}^s (\boldsymbol{\Psi}_{ie}^n)^T \left(\boldsymbol{\Phi}\hat{\boldsymbol{w}}_{P,e}^n - \boldsymbol{f}_e^n\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{n-1} + \Delta t \sum_{j=1}^s a_{ej} \boldsymbol{\Phi}\hat{\boldsymbol{w}}_{P,j}^n\right)\right)\right\|_2.$$

The explicit state updates from (2.7) and (3.11) and the bound for ζ yields

$$\|\delta \boldsymbol{x}_P^n\|_2 \leq \left(1 + \kappa \Delta t \sum_{k=1}^s |b_k| \sum_{i=1}^s [\boldsymbol{D}^{-1}]_{ki}\right) \|\delta \boldsymbol{x}_P^{n-1}\|_2$$
$$+ \Delta t \sum_{k=1}^s |b_k| \sum_{i=1}^s [\boldsymbol{D}^{-1}]_{ki} \left\|(\boldsymbol{I} - \mathbb{P}_i^n)\boldsymbol{f}_i^n\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{n-1} + \Delta t \sum_{j=1}^s a_{ij} \boldsymbol{\Phi}\hat{\boldsymbol{w}}_{P,j}^n\right)\right\|_2$$
$$+ \Delta t \sum_{k=1}^s |b_k| \sum_{i=1}^s [\boldsymbol{D}^{-1}]_{ki} \cdot$$
$$\left\| \boldsymbol{\Phi}\left((\boldsymbol{\Psi}_{ii}^n)^T \boldsymbol{\Phi}\right)^{-1} \sum_{e=1, e\neq i}^s (\boldsymbol{\Psi}_{ie}^n)^T \left(\boldsymbol{\Phi}\hat{\boldsymbol{w}}_{P,e}^n - \boldsymbol{f}_e^n\left(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P^{n-1} + \Delta t \sum_{j=1}^s a_{ej} \boldsymbol{\Phi}\hat{\boldsymbol{w}}_{P,j}^n\right)\right)\right\|_2.$$

An induction argument yields the bound (6.29). □

As with linear multistep schemes, the error bound for the Galerkin ROM in (6.28) depends on the orthogonal projection error of $\boldsymbol{f}_i^n$ onto range $(\boldsymbol{\Phi})$, while the LSPG ROM error bound depends on an oblique projection; however, because the oblique projector depends on the LSPG ROM solution, this bound can be smaller than the Galerkin bound (as was demonstrated in Corollary 6.4—note that Backward Euler is also a Runge–Kutta scheme). Further, notice that the complex 'path traversing' that appears in the linear multistep error bounds is not present for Runge–Kutta schemes; this is a consequence of the fact that previous time steps only influence the error at the current time step through induction for Runge–Kutta schemes. In addition, note that the LSPG ROM error bound is more complex than the Galerkin bound; the final line in bound (6.29) is a consequence of the test-basis coupling (4.8) for general implicit Runge–Kutta schemes. Finally, we point out that both bounds grow exponentially in time due to the amplification factor. We now present a simpler version of this error bound for explicit Runge–Kutta and DIRK schemes.

Corollary 6.8 (*a posteriori* error bounds: explicit Runge–Kutta and DIRK). *Under the assumptions of Theorem 6.7 for explicit RK ($\theta = 1$) and DIRK ($\theta = 0$) schemes, we have*

$$\|\delta \boldsymbol{x}_P^n\|_2 \leq \Delta t \sum_{\ell=0}^{n-1} \Big(1 + \kappa \Delta t \sum_{k=1}^{s} |b_k| \sum_{i=1}^{s} [\boldsymbol{D}^{-1}]_{ki}\Big)^{\ell} \cdot \\ \Bigg(\sum_{k=1}^{s} |b_k| \sum_{i=1}^{s} [\boldsymbol{D}^{-1}]_{ki} \Big\| (\boldsymbol{I} - \mathbb{P}_i^{n-\ell}) \boldsymbol{f}_i^{n-\ell} \Big(\boldsymbol{x}_0 + \boldsymbol{\Phi} \hat{\boldsymbol{x}}_P^{n-\ell-1} + \Delta t \sum_{j=1}^{i-\theta} a_{ij} \boldsymbol{\Phi} \hat{\boldsymbol{w}}_{P,j}^{n-\ell} \Big) \Big\|_2 \Bigg). \tag{6.30}$$

Proof. For explicit and diagonally implicit Runge–Kutta schemes (4.12), we have $\boldsymbol{\Psi}_{ij}^n = 0$ when $i \neq j$. The proof is then an immediate consequence of (6.29). □

Note that the LSPG error bound (6.30) resembles the Galerkin error bound (6.28) much more closely than the previous LSPG bound (6.29), as explicit Runge–Kutta and DIRK schemes remove the coupling of the test basis across stages. We are now ready to state the *a priori* versions of the Galerkin Runge–Kutta schemes (6.28) and the LSPG Runge–Kutta schemes (6.29).

Theorem 6.9 (*a priori* error bounds: Runge–Kutta schemes). *If ($\mathbf{A_1}$) holds and Δt is such that*

(*a*) *the matrices $\boldsymbol{D} \in \mathbb{R}^{s \times s}$ (defined in Theorem 6.7) and $\bar{\boldsymbol{D}}^n \in \mathbb{R}^{s \times s}$ with entries $[\bar{\boldsymbol{D}}^n]_{ij} := \delta_{ij} - \kappa \Delta t |a_{ij}| \|\mathbb{P}_i^n\|$ are invertible, and*

(*b*) *for every $\boldsymbol{x}, \boldsymbol{y} \geq 0$, if $\boldsymbol{D}\boldsymbol{x} \leq \boldsymbol{y}$ then $\boldsymbol{x} \leq \boldsymbol{D}^{-1}\boldsymbol{y}$ and if $\bar{\boldsymbol{D}}^m \boldsymbol{x} \leq \boldsymbol{y}$ then $\boldsymbol{x} \leq [\bar{\boldsymbol{D}}^m]^{-1}\boldsymbol{y}$, $m \in \mathbb{N}(n)$,*

then

$$\|\delta \boldsymbol{x}_G^n\|_2 \leq \Delta t \sum_{\ell=0}^{n-1} \Big(1 + \kappa \Delta t \sum_{k=1}^{s} |b_k| \sum_{i=1}^{s} [\boldsymbol{D}^{-1}]_{ki}\Big)^{\ell} \cdot \\ \Big(\sum_{k=1}^{s} |b_k| \sum_{i=1}^{s} [\boldsymbol{D}^{-1}]_{ki} \Big\| (\boldsymbol{I} - \mathbb{V}) \boldsymbol{f}_i^{n-\ell} \Big(\boldsymbol{x}_0 + \boldsymbol{x}_\star^{n-\ell-1} + \Delta t \sum_{j=1}^{s} a_{ij} \boldsymbol{w}_{\star,j}^{n-\ell} \Big) \Big\|_2 \Big). \tag{6.31}$$

and

$$\|\delta \boldsymbol{x}_P^n\|_2 \leq \Delta t \sum_{\ell=0}^{n-1} \prod_{m=0}^{\ell-1} \Big(1 + \kappa \Delta t \sum_{k=1}^{s} |b_k| \sum_{i=1}^{s} [[\bar{\boldsymbol{D}}^{n-m}]^{-1}]_{ki}\Big) \cdot \\ \Bigg(\sum_{k=1}^{s} |b_k| \sum_{i=1}^{s} [[\bar{\boldsymbol{D}}^{n-\ell}]^{-1}]_{ki} \Big\| (\boldsymbol{I} - \mathbb{P}_i^{n-\ell}) \boldsymbol{f}_i^{n-\ell} \Big(\boldsymbol{x}_0 + \boldsymbol{x}_\star^{n-\ell-1} + \Delta t \sum_{j=1}^{s} a_{ij} \boldsymbol{w}_{\star,j}^{n-\ell} \Big) \Big\|_2 \\ + \sum_{k=1}^{s} |b_k| \sum_{i=1}^{s} [[\bar{\boldsymbol{D}}^{n-\ell}]^{-1}]_{ki} \cdot \\ \Big\| \boldsymbol{\Phi} \Big((\boldsymbol{\Psi}_{ii}^{n-\ell})^T \boldsymbol{\Phi} \Big)^{-1} \sum_{e=1, e \neq i}^{s} (\boldsymbol{\Psi}_{ie}^{n-\ell})^T \Big(\boldsymbol{\Phi} \hat{\boldsymbol{w}}_{P,e}^{n-\ell} - \boldsymbol{f}_e^{n-\ell} \Big(\boldsymbol{x}_0 + \boldsymbol{\Phi} \hat{\boldsymbol{x}}_P^{n-\ell-1} + \Delta t \sum_{j=1}^{s} a_{ej} \boldsymbol{\Phi} \hat{\boldsymbol{w}}_{P,j}^{n-\ell} \Big) \Big) \Big\|_2 \Bigg), \tag{6.32}$$

where we have used the convention that the empty product is equal to one.

Proof. The proof of (6.31) follows the Galerkin-ROM derivation in Theorem 6.7, wherein the quantity $\mathbb{V}\boldsymbol{f}_i^n \Big(\boldsymbol{x}_0 + \boldsymbol{x}_\star^{n-1} + \Delta t \sum_{j=1}^{s} a_{ij} \boldsymbol{w}_{\star,j}^n \Big)$ is added and subtracted rather than $\boldsymbol{f}_i^n \Big(\boldsymbol{x}_0 + \boldsymbol{\Phi} \hat{\boldsymbol{x}}_G^{n-1} + \Delta t \sum_{j=1}^{s} a_{ij} \boldsymbol{\Phi} \hat{\boldsymbol{w}}_{G,j}^n \Big)$ and $\|\mathbb{V}\|_2 = 1$ is used. On the other hand, (6.32) follows the LSPG-ROM derivation by adding and subtracting $\mathbb{P}_i^n \boldsymbol{f}_i^n \Big(\boldsymbol{x}_0 + \boldsymbol{x}_\star^{n-1} + \Delta t \sum_{j=1}^{s} a_{ij} \boldsymbol{w}_{\star,j}^n \Big)$ instead of $\boldsymbol{f}_i^n \Big(\boldsymbol{x}_0 + \boldsymbol{\Phi} \hat{\boldsymbol{x}}_P^{n-1} + \Delta t \sum_{j=1}^{s} a_{ij} \boldsymbol{\Phi} \hat{\boldsymbol{w}}_{P,j}^n \Big)$. □

As with linear multistep schemes, the rightmost term in the Galerkin *a priori* bound will always be smaller than that for the LSPG bound, as the former associates with an orthogonal projection error of a

fixed vector. In addition, the LSPG bound depends on the LSPG ROM solution; while this dependence could be removed, the bound in its current form facilitates comparison with the Galerkin bound. We again notice the complex structure of the estimator in (6.32) compared to (6.31). To better understand the behavior the LSPG estimator in (6.32) we consider two subcases: explicit Runge–Kutta and DIRK schemes.

Corollary 6.10 (*a priori* error bounds: explicit Runge–Kutta and DIRK). *Under the assumptions of Theorem 6.9 for explicit RK ($\theta = 1$) and DIRK ($\theta = 0$) schemes, we have*

$$\begin{aligned}\|\delta \boldsymbol{x}_P^n\|_2 \leq \Delta t \sum_{\ell=0}^{n-1} \prod_{m=0}^{\ell-1} \Big(1 + \kappa \Delta t \sum_{k=1}^{s} |b_k| \sum_{i=1}^{s} [[\bar{\boldsymbol{D}}^{n-m}]^{-1}]_{ki}\Big) \cdot \\ \left(\sum_{k=1}^{s} |b_k| \sum_{i=1}^{s} [[\bar{\boldsymbol{D}}^{n-\ell}]^{-1}]_{ki} \Big\| (\boldsymbol{I} - \mathbb{P}_i^{n-\ell}) \boldsymbol{f}_i^{n-\ell}\Big(\boldsymbol{x}_0 + \boldsymbol{x}_\star^{n-\ell-1} + \Delta t \sum_{j=1}^{i-\theta} a_{ij} \boldsymbol{w}_{\star,j}^{n-\ell}\Big)\Big\|_2 \right).\end{aligned} \tag{6.33}$$

Proof. For explicit and diagonally implicit Runge–Kutta schemes (4.12), we have $\boldsymbol{\Psi}_{ij}^n = 0$ when $i \neq j$. The proof is then an immediate consequence of (6.32). □

Owing to the fact that the explicit Runge–Kutta and DIRK schemes removes the coupling of the basis, we again notice that the LSPG error bound (6.33) resembles the Galerkin ROM error bound (6.31).

7. Numerical experiments

This section compares the performance of Galerkin and LSPG ROMs on a computational-fluid-dynamics (CFD) application using a basis constructed by proper orthogonal decomposition. These experiments highlight the importance of the previous analyses, in particular the limiting equivalence of Galerkin and LSPG ROMs (Theorem 5.3), superior accuracy of the LSPG ROM compared with the Galerkin ROM (Corollary 6.4), and performance improvement of the LSPG ROM when an intermediate time step is selected (Corollary 6.5 and Remark 6.6).

Note that these experiments could be carried out on any dynamical system yielding a system of nonlinear ODEs (2.1); we have selected compressible turbulent fluid dynamics due to both its wide interest and challenging nature: limited progress has been made to date on developing robust, accurate ROMs for such problems. The numerical experiments highlight this fact, as standard Galerkin ROMs generate unstable responses in all cases.

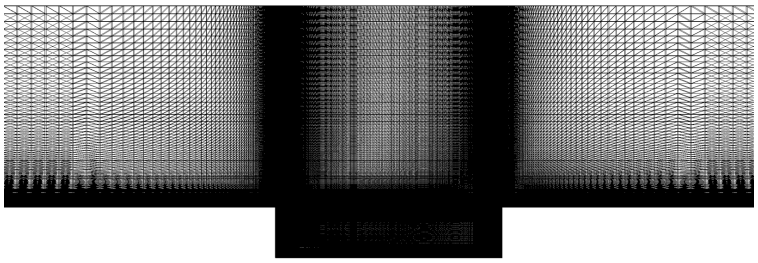
7.1. Problem description

The Galerkin and LSPG ROMs are implemented in AERO-F [34, 30], a massively parallel compressible-flow solver. AERO-F solves the steady or unsteady compressible Navier–Stokes equations with various closure models available for turbulent flow, and employs a second-order node-centered finite-volume scheme. For model-reduction algorithms, all linear least-squares problems and singular value decompositions are computed in parallel using the ScaLAPACK library [15].

The full-order model corresponds to an unsteady Navier–Stokes simulation of a two-dimensional open cavity using AERO-F's DES turbulence model (based on the Spalart–Allmaras one-equation model [54]) and a wall-function boundary condition applied on solid surface boundaries. The fluid domain is discretized by a mesh with 192,816 nodes and 573,840 tetrahedra (Figure 2). The two-dimensional geometry is discretized in three dimensions by considering a slab of thin, but finite thickness, in the z-direction; the resulting grid is one element wide and is created by extruding a distance that is 1% of the cavity length. The viscosity is assumed to be constant, and the Reynolds number based on cavity length is 6.30×10^6, while the free-stream Mach number is 0.6. Due to the turbulence model and three-dimensional domain, the number of conservation equations per node is 6, and therefore the dimension of the CFD model is $N = 1,156,896$. Roe's scheme is employed to discretize the convective fluxes, and a linear variation of the solution is assumed within each control volume, which leads to a second-order space-accurate scheme. The viscous flux is discretized using a centered Galerkin scheme. We employ a low-numerical-dissipation scheme that gives fifth-order formal order of accuracy (with uniform mesh spacing) on inviscid, one-dimensional problems.

Flow simulations are performed within a time interval $t \in [0, T]$ with $T = 12.5$ seconds. We employ the second-order accurate implicit three-point backward difference scheme, which is a linear multistep scheme

(a) Full domain



(b) Detail around cavity

Figure 2: Computational mesh: $x - y$ plane cut.

characterized by $k = 2$, $\alpha_0 = 1$, $\alpha_1 = -4/3$, $\alpha_2 = 1/3$, $\beta_0 = 2/3$, $\beta_1 = \beta_2 = 0$, for time integration; future work will perform numerical experiments with Runge–Kutta schemes. The OΔE (2.3) arising at each time step is solved by a Newton–Krylov method, where GMRES is employed as the iterative linear solver with a restrictive additive Schwarz preconditioner (with no fill in) and the previous 50 Krylov vectors are employed for orthogonalization. Convergence is declared when the residual norm is reduced to a factor of 10^{-3} of its starting value. All flow computations are performed in a non-dimensional setting.

The initial condition $\boldsymbol{x}_0$ is provided by first computing a steady-state solution, and using that solution as an initial guess for an unsteady 'transient' simulation (which captures the initial transient before the flow reaches a quasi-periodic state) of 7.5 seconds. The state at the end of the unsteady transient simulation is then used as the initial condition for the subsequent simulations. The steady-state calculation is characterized by the same parameters as above, except that it employs local time stepping with a maximum CFL number of 100, it uses the first-order implicit backward Euler time integration scheme, it assumes a linear variation of the solution within each control volume, it employs a Spalart–Allmaras turbulence model, and it employs only one Newton iteration per (pseudo) time step.

The output of interest is the pressure at location (0.0001,-0.0508,0.0025), which is shown in the bottom row of Figure 4. All errors are reported as the ℓ^2 relative error in this quantity, i.e.,

$$\varepsilon(p, p_\star) = \frac{\sqrt{\sum_{n=1}^{T/\Delta t_\star} \left(P_\star(p)(n\Delta t_\star) - p_\star(n\Delta t_\star)\right)^2}}{\sqrt{\sum_{n=1}^{T/\Delta t} p_\star(n\Delta t_\star)^2}},$$

where $p : \mathbb{N}(T/\Delta t) \to \mathbb{R}$ is the pressure for the model of interest, $p_\star : \mathbb{N}(T/\Delta t_\star) \to \mathbb{R}$ is this pressure response of the designated 'truth' model (typically the full-order model), and $P_\star$ is a linear interpolation of the pressure response onto the grid based on the truth-model time step $\Delta t_\star$.

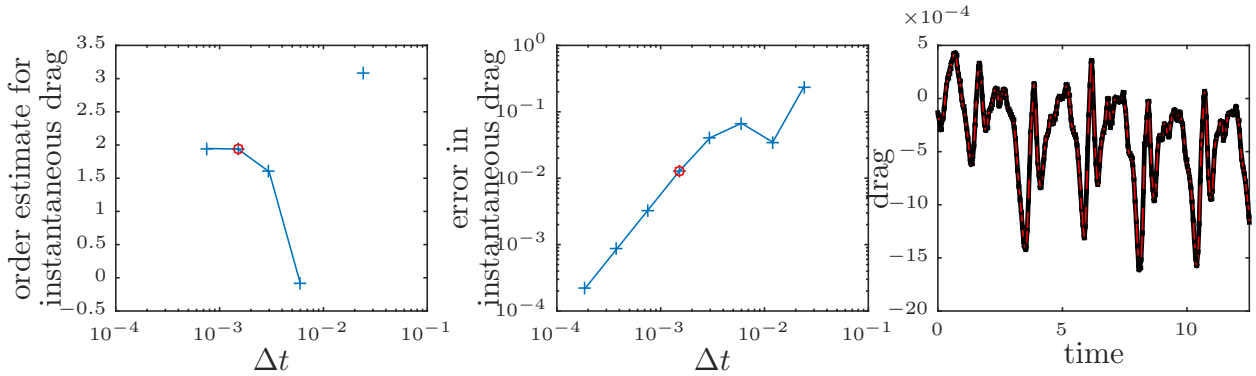
All computations are performed in double-precision arithmetic on a parallel Linux cluster[3] using 48 cores across 6 nodes.

7.2. Time-step verification

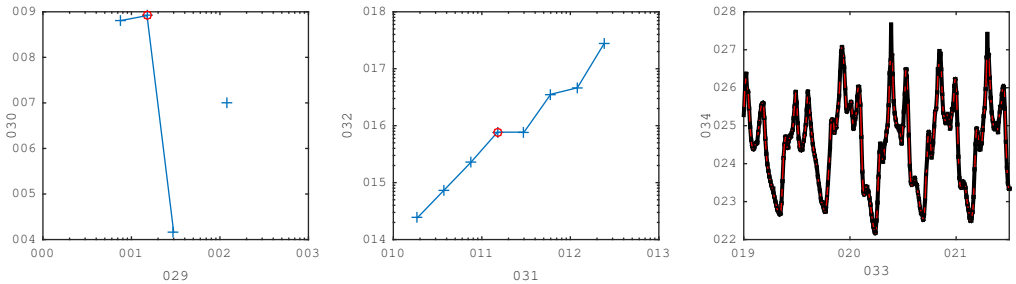
Because this paper considers the time step to be an important parameter in model reduction, we first perform a time-step verification study to ensure we employ an appropriate 'nominal' time step. Figure

[3]The cluster contains 8-core compute nodes that each contain a 2.93 GHz dual socket/quad core Nehalem X5570 processor with 12 GB of memory. The interconnect is a 3D torus InfiniBand.

3 reports these results using a time-step refinement factor of two. A time step of $\Delta t_\star = 0.0015$ seconds yields observed convergence rates in both the instantaneous drag force on the lower wall and instantaneous pressure at $t = T$ that are close to the asymptotic rate of convergence (2.0) of three-point BDF2 scheme. Further, this value also leads to sub-2% errors in both quantities, which we deem to be sufficient for this set of experiments.



(a) Drag: $\Delta t_\star = 0.0015$ yields an approximate rate of convergence of 1.94 and an estimated error in the output quantity (computed via Richardson extrapolation) of 1.26×10^{-2}. The red points denote the result for $\Delta t_\star = 0.0015$. The rightmost plot shows the time-dependent response for the finest tested time step $\Delta t = 1.875 \times 10^{-4}$ (black solid) and the converged time step $\Delta t_\star = 0.0015$ (red dashed).



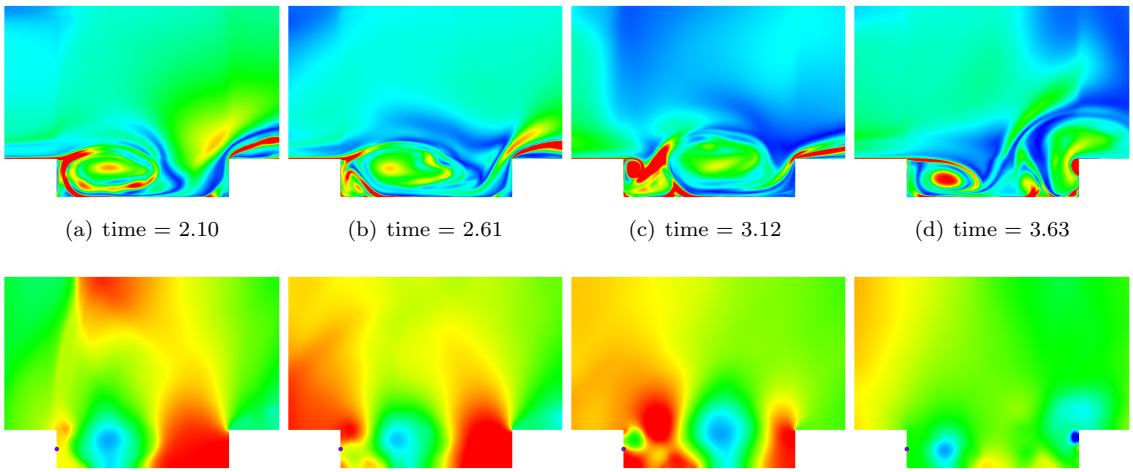
(b) Pressure: $\Delta t_\star = 0.0015$ yields an approximate rate of convergence of 1.83 and an estimated error in the output quantity (computed via Richardson extrapolation) of 7.68×10^{-4}. The red points denote the result for $\Delta t_\star = 0.0015$. The rightmost plot shows the time-dependent response for the finest tested time step $\Delta t = 1.875 \times 10^{-4}$ (black solid) and the converged time step $\Delta t_\star = 0.0015$ (red dashed).

Figure 3: Time-step verification study. Note that the approximated convergence rates are close to the asymptotic value of 2.0 for the BDF2 scheme.

Figure 4 shows several instantaneous snapshots of the vorticity field and corresponding pressure field generated by the high-fidelity CFD model. The flow within the cavity is quasi-periodic; during one cycle, vorticity is shed from the leading edge of the cavity, convects downstream, and impinges on the aft edge of the cavity. Upon impingement, an acoustic disturbance is generated which propagates upstream and scatters on the leading edge of the cavity, generating a new vortical disturbance to initiate the next oscillation cycle. The pressure fields in the bottom row of Figure 4 reveal regions of low pressure (blue contours) associated with vortices, as well as acoustic disturbances both within the cavity and radiating outside the cavity. This complex flow is governed by the interactions of several nonlinear processes, including roll-up of the shear layer vortices, impingement of the vortices on the aft wall resulting in sound generation, propagation of nonlinear acoustic waves, and interaction of these waves with the shear layer vorticity.

7.3. Reduced-order models

To construct both the Galerkin and LSPG ROMs, we employ the proper orthogonal decomposition (POD) technique; we employ a constant weighting matrix $\boldsymbol{A} = \boldsymbol{I}$ for the LSPG ROM. To construct the POD basis, we set $\boldsymbol{\Phi} \leftarrow \boldsymbol{\Phi}(\mathcal{X}, \nu)$, where $\boldsymbol{\Phi}$ is computed via Algorithm 1 of the appendix with snapshots consisting of the initial-condition-centered full-order model states $\mathcal{X} = \{\boldsymbol{x}_\star(k\Delta t_\star) - \boldsymbol{x}_0\}_{k=1}^{8334}$, where $\boldsymbol{x}_\star$ denotes the FOM

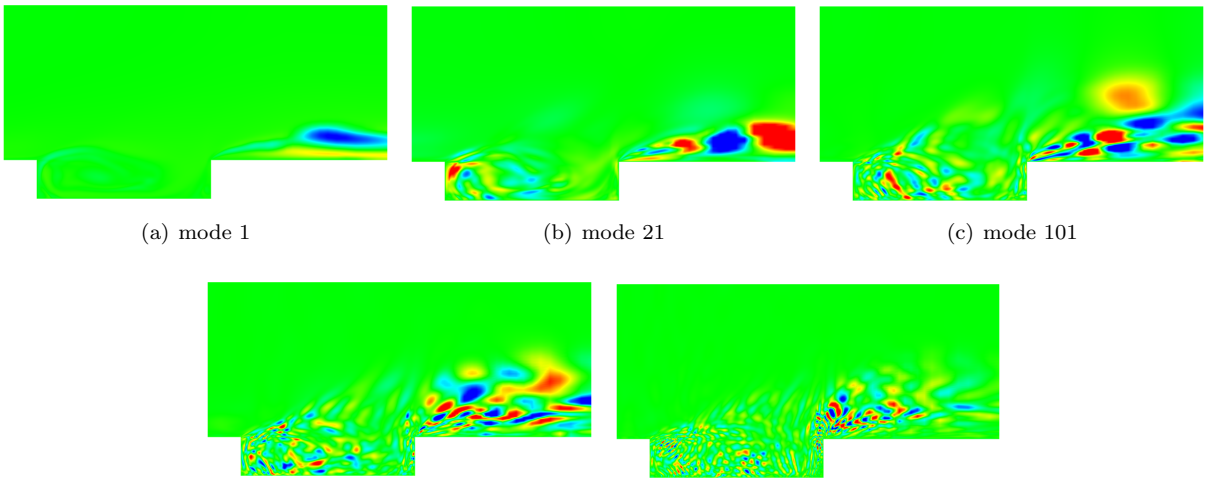


(a) time = 2.10 (b) time = 2.61 (c) time = 3.12 (d) time = 3.63

(e) time = 2.10 (f) time = 2.61 (g) time = 3.12 (h) time = 3.63

Figure 4: Instantaneous CFD vorticity field (top) and pressure field (bottom) during one oscillation cycle. The dot on the forward wall of the cavity indicates the location of the pressure signal output.

response computed for a time step of $\Delta t_\star = 0.0015$. Three values of the energy criterion $\nu \in [0, 1]$ are used during the experiments: $\nu = 1 - 10^{-4}$ ($p = 204$), $\nu = 1 - 10^{-5}$ ($p = 368$), and $\nu = 1 - 10^{-6}$ ($p = 564$). Figure 5 shows a selection of the energy component of the computed POD modes. Note that as the mode number increases, the modes capture finer spatial-scale behavior, which we expect to be associated with finer time-scale behavior; this will be verified in Section 7.5.1.

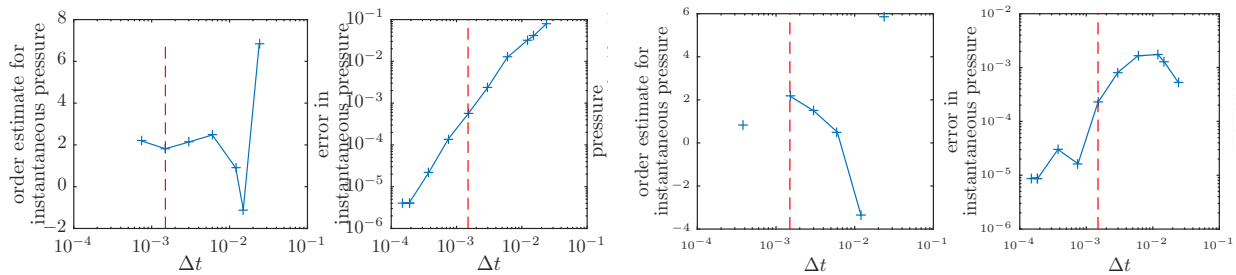


(a) mode 1 (b) mode 21 (c) mode 101

(d) mode 201 (e) mode 401

Figure 5: Visualization of the energy component of the POD modes.

We first repeat the time-step verification study, but we do so for the reduced-order models (again using the BDF2 scheme) in the time interval $0 \leq t \leq 0.55$, as all Galerkin ROMs remain stable in this time interval. Figure 6 reports these results. First, we note that the Galerkin ROM converges an approximated rate of 2.0, which is what we expect given that the Galerkin ROM simply associates with a time-step-independent ODE (3.2). However, the LSPG ROM does not exhibit this behavior; in fact the error convergence is not even monotonic. This is likely due to the fact that the method does not associate with a time-step-independent ODE.



(a) Galerkin reduced-order model (b) LSPG reduced-order model

Figure 6: Time-step verification study for Galerkin and LSPG reduced-order models for $p = 368$ and $0 \leq t \leq 0.55$. While the approximated convergence rate for the Galerkin reduced-order model is close to the asymptotic value of 2.0 for the BDF2 scheme, this is not observed for the LSPG reduced-order model. Note that the error for each ROM is computed with respect its response for finest Richardson extrapolation, and a dashed red line indicates the snapshot-collection time step $\Delta t_\star = 0.0015$. See Figure 7(c–d) for the associated time-dependent responses.

We next perform simulations for both reduced-order models for all tested basis dimensions and time steps; Figure 7 reports the time-dependent responses. When a response stops before the end of the time interval, this indicates that a negative pressure was encountered, which causes AERO-F to exit the simulation. We interpret this phenomenon as a non-physical instability.

First, note that the Galerkin ROMs become unstable (i.e., generate a negative pressure) for all time steps and all basis dimensions. This is consistent with previously reported results [20, 21, 19, 18] that indicate Galerkin projection almost always leads to inaccurate responses for compressible fluid-dynamics problems. In contrast, the LSPG ROM results in many stable, accurate responses for all basis dimensions. Further, LSPG responses exhibit a clear dependence on the time step Δt. Subsequent sections provide a deeper analysis of this dependence.

7.4. Limiting case: comparison

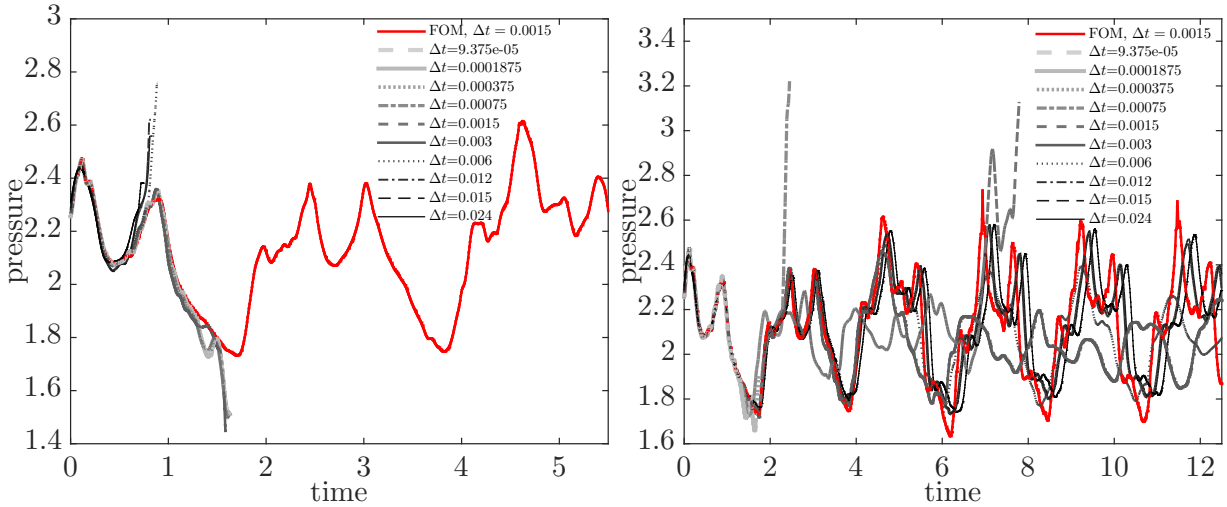
We next compare the responses of the Galerkin and LSPG ROMs for small time windows (when the Galerkin responses remain stable) and small time steps. Figure 8 reports $\varepsilon(p_{\text{discrete opt.}}, p_{\text{Gal}_\star})$—which is the difference between the pressure responses generated by the LSPG ROM with different time steps and the Galerkin ROM with a fixed time step $\Delta t = 1.875 \times 10^{-4}$ (the smallest tested time step)—for a time window $0 \leq t \leq 1.1$. These responses support an important conclusion (see Theorem 5.3): the Galerkin and LSPG ROMs are equal in the limit of $\Delta t \to 0$.[4] This has significant consequences for the LSPG ROM, as decreasing the time step leads to the same *unstable* response as Galerkin; larger time steps are needed to ensure the LSPG ROM is stable for the entire time interval.

Figure 9 reports $\varepsilon(p_{\text{discrete opt.}}, p_{\text{FOM}_\star})$ and $\varepsilon(p_{\text{Gal.}}, p_{\text{FOM}_\star})$—which are the differences between the two ROM-generated pressure responses and the full-order model pressure response for $\Delta t = 1.875 \times 10^{-4}$—as a function of the time step for all three basis dimensions and three time intervals. These results highlight a critical observation: the LSPG ROM is *more accurate* for an intermediate time step. This not only supports the result of Corollary 6.5, but provides an interesting insight: taking a larger time step not only leads to better speedups (i.e., the end of the time interval is reached in fewer time steps), but it also decreases the error, sometimes significantly. This is further explored in the next section.

7.5. Time-step selection

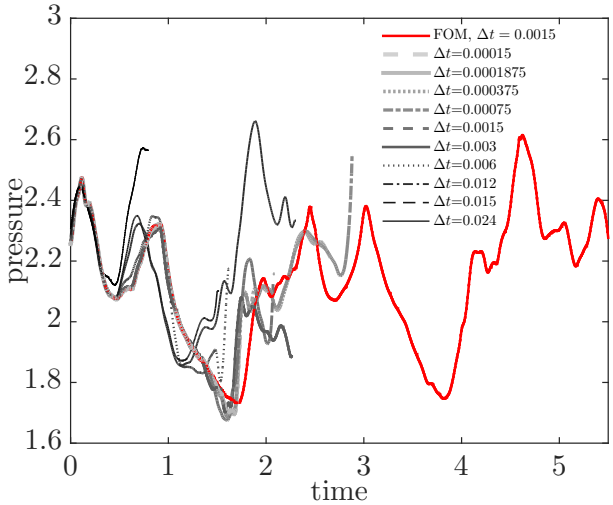
Recall from Corollary 6.5 and Remark 6.6 that decreasing the time step Δt has a non-obvious effect on the error bound for the LSPG ROM. We now assess these effects for the current problem.

[4]Note that in the $p = 564$ case, it is not clear if the difference is converging to zero. This is likely due to the fact that the time steps are not sufficiently small to detect convergence to zero in this case. In fact, as the basis dimension p increases, the basis captures finer temporal behavior (as will be shown in Figure 10) and so the time scale of the ROM response will be smaller; in turn, smaller time steps Δt will be required to detect convergent behavior.

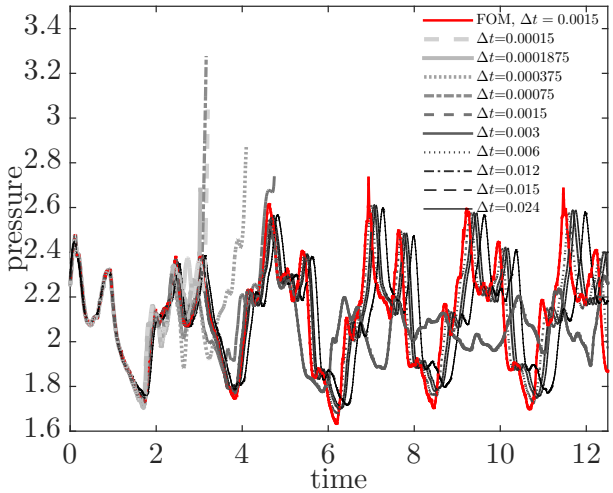


(a) Galerkin, $p = 204$

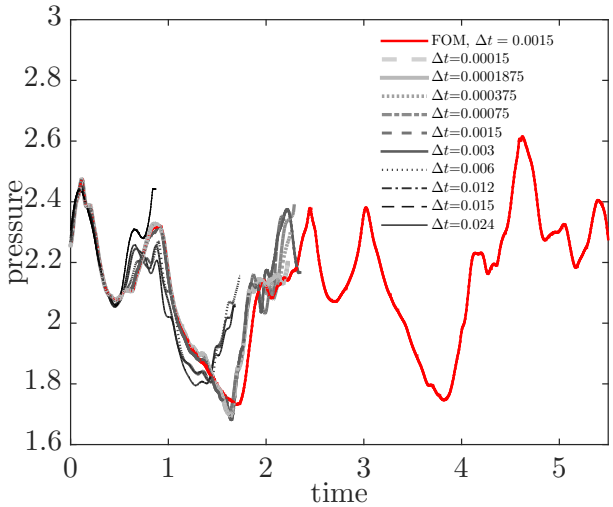
(b) LSPG, $p = 204$



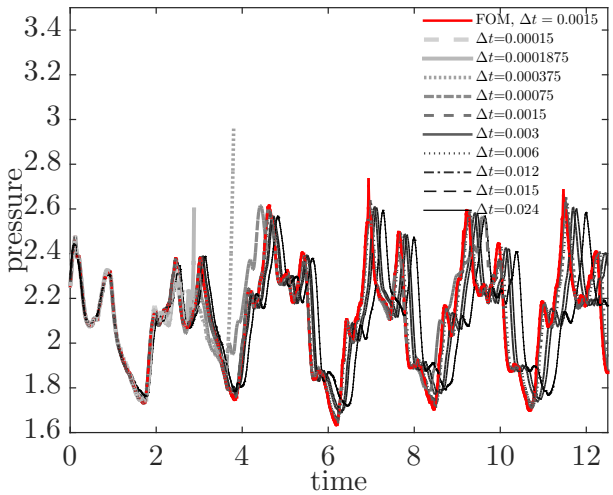
(c) Galerkin, $p = 368$



(d) LSPG, $p = 368$

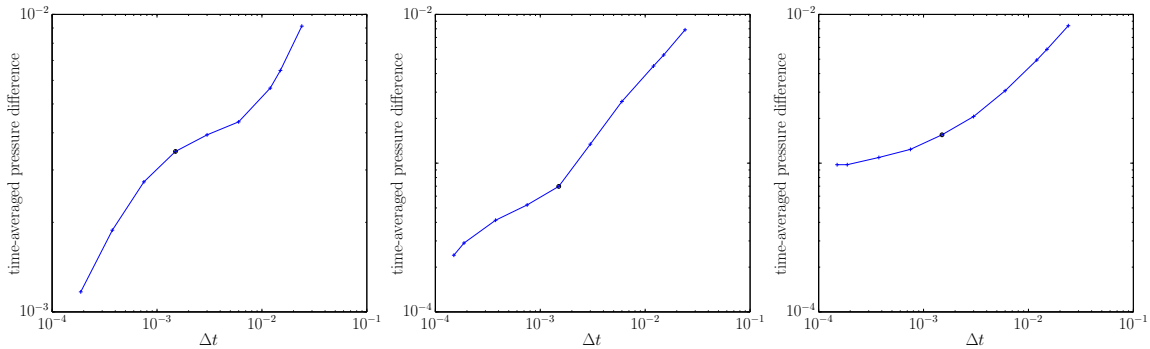


(e) Galerkin, $p = 564$



(f) LSPG, $p = 564$

Figure 7: Responses generated by Galerkin and LSPG ROMs for different basis sizes p and time steps Δt



(a) $p = 204$ (b) $p = 368$ (c) $p = 564$

Figure 8: Difference $\varepsilon(p_{\text{discrete opt.}}, p_{\text{Gal}_\star})$ between the pressure responses generated by the LSPG ROM with different time steps and the Galerkin ROM with a fixed time step $\Delta t = 1.875 \times 10^{-4}$ in $0 \leq t \leq 1.1$. This demonstrates convergence of the LSPG ROM to Galerkin as $\Delta t \to 0$.

7.5.1. Spectral content of POD basis

In our interpretation of the error bound (6.23) for the LSPG ROM applied to the backward Euler scheme, we noted that the time step should be 'matched' to the spectral content of the trial basis $\boldsymbol{\Phi}$. This is of practical importance, as selecting an appropriate time step for the ROM should take into account the relevant temporal dynamics associated with the basis. For example, a time step may be too small if the basis has filtered out modes with a time scale matching that of the time step. If we assume that the basis $\boldsymbol{\Phi}$ is computed via POD, then we would expect the vectors to be naturally ordered such that lower mode numbers are associated with lower temporal frequencies. Then, including additional modes has the effect of encoding information at higher frequencies. It follows that the time step should be decreased as additional modes are retained in construction of the ROM.

Here we investigate the validity of this assumption by examining the spectral content of the POD basis vectors for the current cavity-flow problem. We compute the time histories of the generalized coordinates by projecting the FOM solution onto the POD basis as $\hat{\boldsymbol{x}}_\star(k\Delta t_\star) := \boldsymbol{\Phi}^T(\boldsymbol{x}_\star(k\Delta t_\star) - \boldsymbol{x}_0)$, $k \in \mathbb{N}(8334)$. We then compute power spectral densities of the generalized coordinates $\hat{\boldsymbol{x}}_\star(t)$. Figure 10(a) shows sample spectra, normalized by the total energy in each signal,[5] for several of the POD modes. The figure shows that energy shifts to higher frequencies as the POD mode number increases, confirming our assumption for this example. This is further quantified by calculating a characteristic time-scale τ_{95} associated with each mode; we define this time scale as the inverse of the frequency below which 95 percent of the energy is captured for that mode. Figure 10(b) plots this time scale versus the mode number, showing a clear trend of decreasing time scale with increasing mode number.

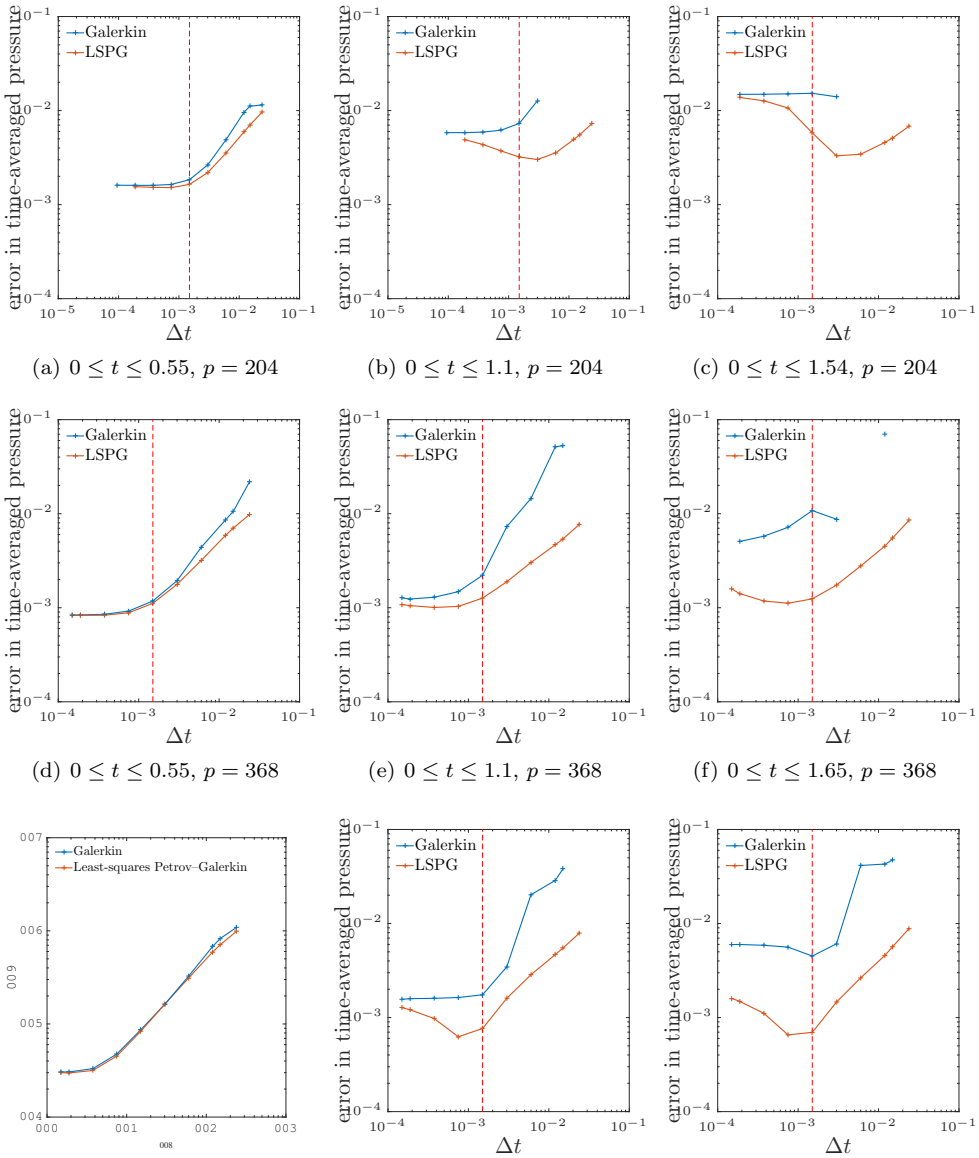
Thus, at least for the present application problem, we expect the optimal time step for the LSPG ROM to decrease as modes are added to the POD basis (this will be verified by Figure 12). Note that systematic calibration could be performed to attempt to automate selection of the ROM time step as a function of basis dimension. While this would be of clear practical interest, we do not pursue it here, as optimal-timestep computation would be complicated in practice by nonlinear interactions arising from the dynamical system, as well as effects from the spatial-discretization error and POD truncation error.

7.5.2. Error bound behavior

Having verified that higher POD mode numbers correspond to smaller wavelengths, we now numerically assess quantities related to the error bound (6.23). First, Figure 11(a) reports the dependence of the maximum relative projection error $\max_k \bar{\mu}_\star^k(\boldsymbol{\Phi}, \Delta t)$ on the time step Δt and the basis dimension, where

$$\bar{\mu}_\star^k(\boldsymbol{\Phi}, \Delta t) := \frac{\|(\boldsymbol{I} - \boldsymbol{\Phi}\boldsymbol{\Phi}^T)(\boldsymbol{x}_\star(k\Delta t) - \boldsymbol{x}_\star((k-1)\Delta t))\|}{\|\boldsymbol{x}_\star(k\Delta t) - \boldsymbol{x}_\star((k-1)\Delta t)\|}.$$

[5]The energy in a time series within some frequency range is obtained by integrating the power spectral density over that range.

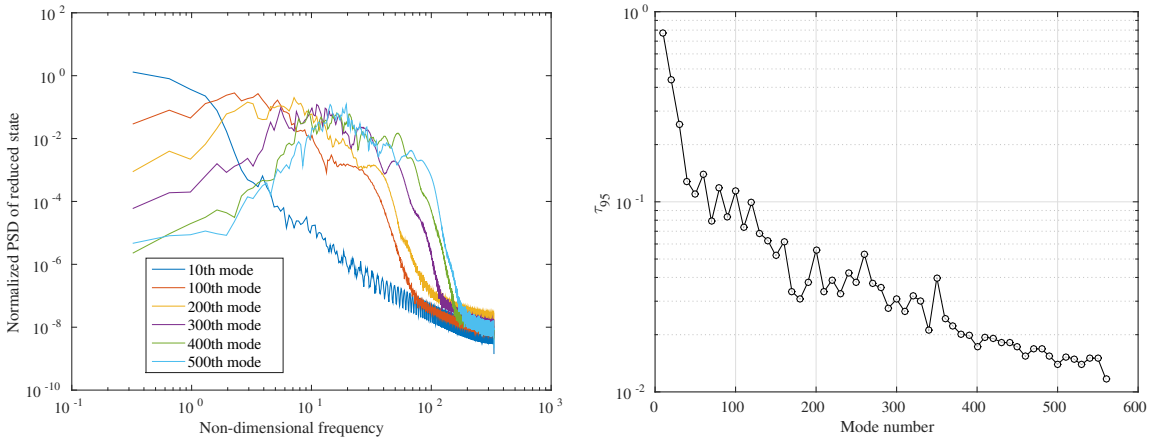


(a) $0 \leq t \leq 0.55$, $p = 204$ (b) $0 \leq t \leq 1.1$, $p = 204$ (c) $0 \leq t \leq 1.54$, $p = 204$

(d) $0 \leq t \leq 0.55$, $p = 368$ (e) $0 \leq t \leq 1.1$, $p = 368$ (f) $0 \leq t \leq 1.65$, $p = 368$

(g) $0 \leq t \leq 0.55$, $p = 564$ (h) $0 \leq t \leq 1.1$, $p = 564$ (i) $0 \leq t \leq 1.65$, $p = 564$

Figure 9: Galerkin errors $\varepsilon(p_{\text{discrete opt.}}, p_{\text{FOM}_\star})$ and Petrov–Galerkin errors $\varepsilon(p_{\text{Gal.}}, p_{\text{FOM}_\star})$ over different time intervals, time steps, and basis dimensions. For reference, a dashed red line indicates the snapshot-collection time step $\Delta t_\star = 0.0015$.



(a) Power spectral densities of several generalized coordinate time series for the cavity-flow problem.

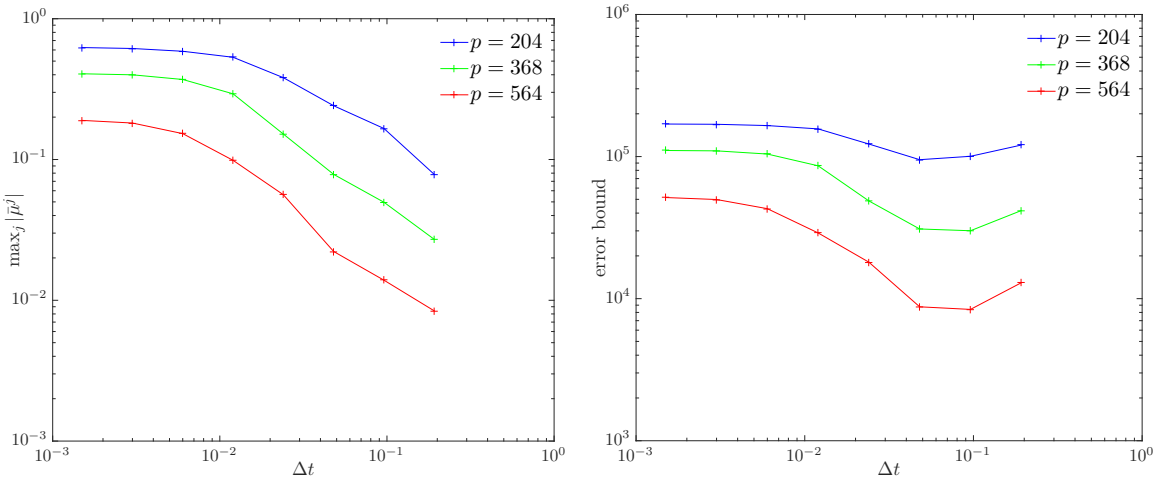
(b) Dependence of the POD mode characteristic time scale on mode number for the cavity-flow problem.

Figure 10: Spectral content of the POD basis.

Note that $\bar{\mu}^k_\star$ is closely related to $\bar{\mu}^k$ from error bound (6.23), as they are equal if $\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{x}}_P(t) = \boldsymbol{x}(t)$ and the LSPG ROM computes $\hat{\boldsymbol{x}}^k_P$ such that $\bar{\mu}^k$ is minimized.

These results confirm that adding basis vectors—which we know has the effect of encoding higher frequency content—significantly reduces the projection error for small time steps Δt, but has less of an effect on larger time steps, as retaining the first POD vectors already enables dynamics at that scale to be captured.

Next, Figure 11(b) plots the error bound (6.23) for a value of $\kappa = 1$ and with $\bar{\mu}^k = \bar{\mu}^k_\star$. This highlights an important result: *selecting an intermediate time step Δt leads to the lowest error bound, regardless of the basis dimension.* Even though this result corresponds to the backward Euler integrator, we expect a similar trend to hold for the present experiment, which uses the BDF2 scheme. The next section assesses the performance of the LSPG ROM, including its dependence on the time step.



(a) Dependence of the maximum relative projection error $\max_k \bar{\mu}^k_\star$ on the time step Δt and basis dimension p

(b) LSPG error bound for the backward Euler scheme (6.23) using experimental data for $\bar{\mu}^k_\star$ and setting $\kappa = 1$ and $\bar{\mu}^k = \bar{\mu}^k_\star$.

Figure 11: Assessment of quantities appearing in error bound (6.23). This analysis suggests that an intermediate time step Δt can reduce errors for the LSPG ROM.

7.6. LSPG ROM performance

We now compare the accuracy and walltime performance of the LSPG ROM as the dimension of the basis, time step, and time interval change. The most salient result from Figure 12 is that choosing an intermediate time step leads to both better accuracy and faster simulation times. This shows that our theoretical analysis of the error bound performed in Section 7.5.2 leads to an actual observed performance improvement. For example, consider the $p = 564$ case over the time interval $0 \leq t \leq 2.5$. In this case, a time step of $\Delta t = 1.875 \times 10^{-4}$ leads to a relative error of 0.0140 and a simulation time of 289 hours; increasing this value to $\Delta t = 1.5 \times 10^{-3}$ reduces the relative error to 9.46×10^{-4} and the simulation time to 35.8 hours, which constitutes roughly an order of magnitude improvement in both quantities. Again, this supports the theoretical results of Corollary 6.5 and highlights the critical importance of the time step for LSPG reduced-order models.

In addition, Figure 12 shows that as the basis dimension increases, the optimal time step decreases; this was anticipated from the spectral analysis performed in Section 7.5.1. In addition, adding POD basis vectors does not improve accuracy for large time steps. We interpret this effect as follows: for larger time steps, the first few POD modes accurately capture 'coarse' phenomena on the scale of the time step. Therefore, accuracy improvement is not achieved by adding modes that encode dynamics that evolve on a time scale finer than the time step itself.

Further, Figure 12(g) highlights that as the basis dimension increases, the error generally decreases, which is an artifact of *a priori* convergence achieved by the LSPG ROM (Remark 4.1). Finally, the figure shows that as the time interval grows, the optimal time step generally increases.

7.7. GNAT: ROM with complexity reduction

In this section, we perform a similar study, but equip the LSPG ROM with complexity reduction in order to achieve computational savings. In particular, we employ the GNAT method [19, 20, 21], which solves Eq. (4.1) with $\boldsymbol{A} = (\boldsymbol{P}\boldsymbol{\Phi}_r)^{+}\boldsymbol{P}$, where $\boldsymbol{\Phi}_r$ is a basis for the residual and $\boldsymbol{P}$ consisting of selected rows of the identity matrix.

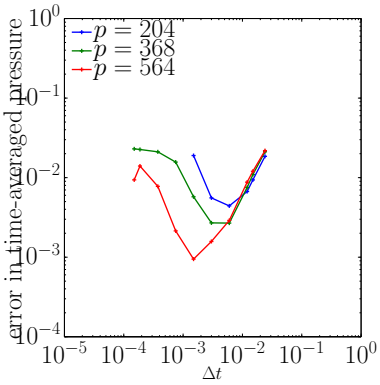
The problem is identical to that described in Section 7.1 except that we take $T = 5.5$ seconds and employ a second-order space-accurate dissipation scheme wherein a linear variation of the solution is assumed within each control volume.[6] For this simulation, the full-order model consumes 5.0 hours on 48 cores across six compute nodes.

To construct the trial basis $\boldsymbol{\Phi}$ and basis for the residual $\boldsymbol{\Phi}_r$ for the GNAT models, we again employ POD. In particular, we set $\boldsymbol{\Phi} \leftarrow \boldsymbol{\Phi}(\mathcal{X}, \nu)$, where $\boldsymbol{\Phi}$ is computed via Algorithm 1 with snapshots consisting of the centered full-order model states $\mathcal{X} = \{\boldsymbol{x}_\star(k\Delta t_\star) - \boldsymbol{x}_0\}_{k=1}^{3668}$. An energy criterion of $\nu = 1 - 10^{-5}$ ($p = 179$) is used during the experiments. For the residual, we employ $\boldsymbol{\Phi}_r \leftarrow \boldsymbol{\Phi}(\mathcal{X}_{\boldsymbol{r}}, \nu_{\boldsymbol{r}})$ via Algorithm 1 with snapshots $\mathcal{X}_{\boldsymbol{r}} = \{\boldsymbol{r}^n(\boldsymbol{x}_0 + \boldsymbol{\Phi}\hat{\boldsymbol{w}}^{n(k)}),\ k \in \mathbb{N}(K(n)),\ n \in \mathbb{N}(2228)\}$ and $\hat{\boldsymbol{w}}^{n(k)}$ corresponding to the LSPG ROM solution at Gauss–Newton iteration k within time step n using a time step of $\Delta t = 6 \times 10^{-3}$. Here, $K(n)$ denotes the number of Newton iterations required for convergence of at time instance n. An energy criterion of $\nu_{\boldsymbol{r}} = 1.0$ is employed. In addition, the GNAT model sets the Jacobian basis equal to residual basis $\boldsymbol{\Phi}_J = \boldsymbol{\Phi}_r$ and employs $n_s = 743$ sample nodes that define $\boldsymbol{P}$, which leads to 4458 rows in $\boldsymbol{P}$ as there are six conservation equations per node due to the turbulence model (see Ref. [21] for definitions).

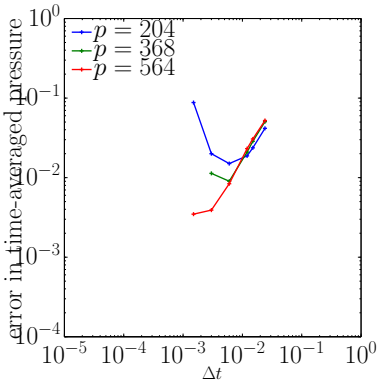
The GNAT implementation in AERO-F is characterized by the sample-mesh concept [21]. Figure 13 depicts the sample mesh for this problem, which was constructed using $n_c = 2228$ working columns [21, Algorithm 3], and includes two layers of nodes around the sample nodes (to enable the residual to be computed at the sample nodes). It is characterized by 7,974 total nodes (4.1% of the original mesh) and 17,070 total volumes (3.0% of the original mesh). Due to the small footprint of the sample mesh, the GNAT simulations are run using only 2 cores on a single compute node.

Figure 14 reports the results obtained with the GNAT ROM using different time steps. Critically, note that the GNAT ROM also exhibits a 'dip' in the optimal time step, with a time step of 6.0×10^{-3} yielding the lowest error. In fact, increasing the time step from 1.5×10^{-3} to 6.0×10^{-3} decreases the error from 3.32% to 2.25% and also significantly increases the computational savings relative to the full-order model

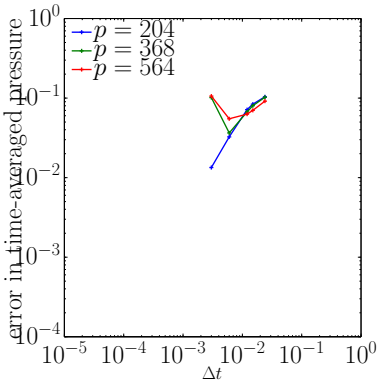
[6] This is done to ensure the sample mesh requires two layers of neighboring nodes for each sample node.



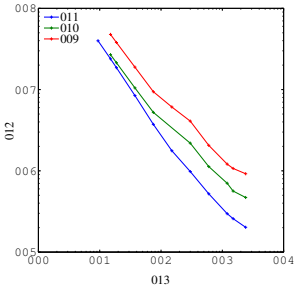
(a) errors for $0 \leq t \leq 2.5$



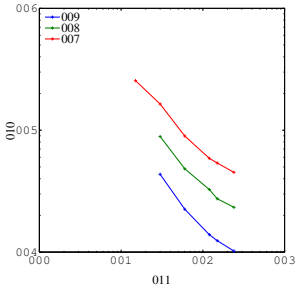
(b) errors for $0 \leq t \leq 5.0$



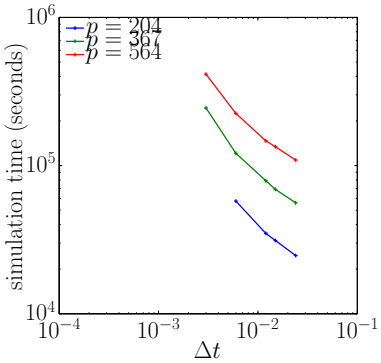
(c) errors for $0 \leq t \leq 12.5$



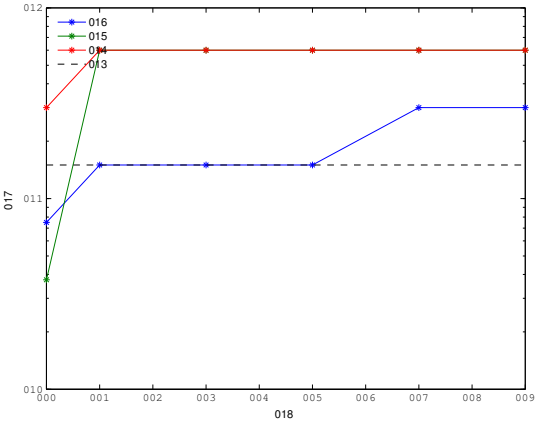
(d) simulation times for $0 \leq t \leq 2.5$



(e) simulation times for $0 \leq t \leq 5.0$



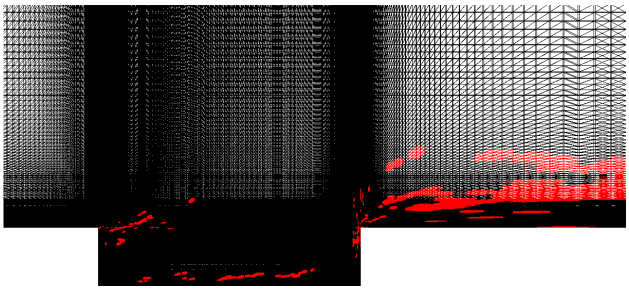
(f) simulation times for $0 \leq t \leq 12.5$



(g) Dependence of optimal time step (in terms of minimizing error) on time interval and basis dimension

Figure 12: Dependence of error and simulation time for the LSPG reduced-order model on the time step Δt, basis dimension, and time interval

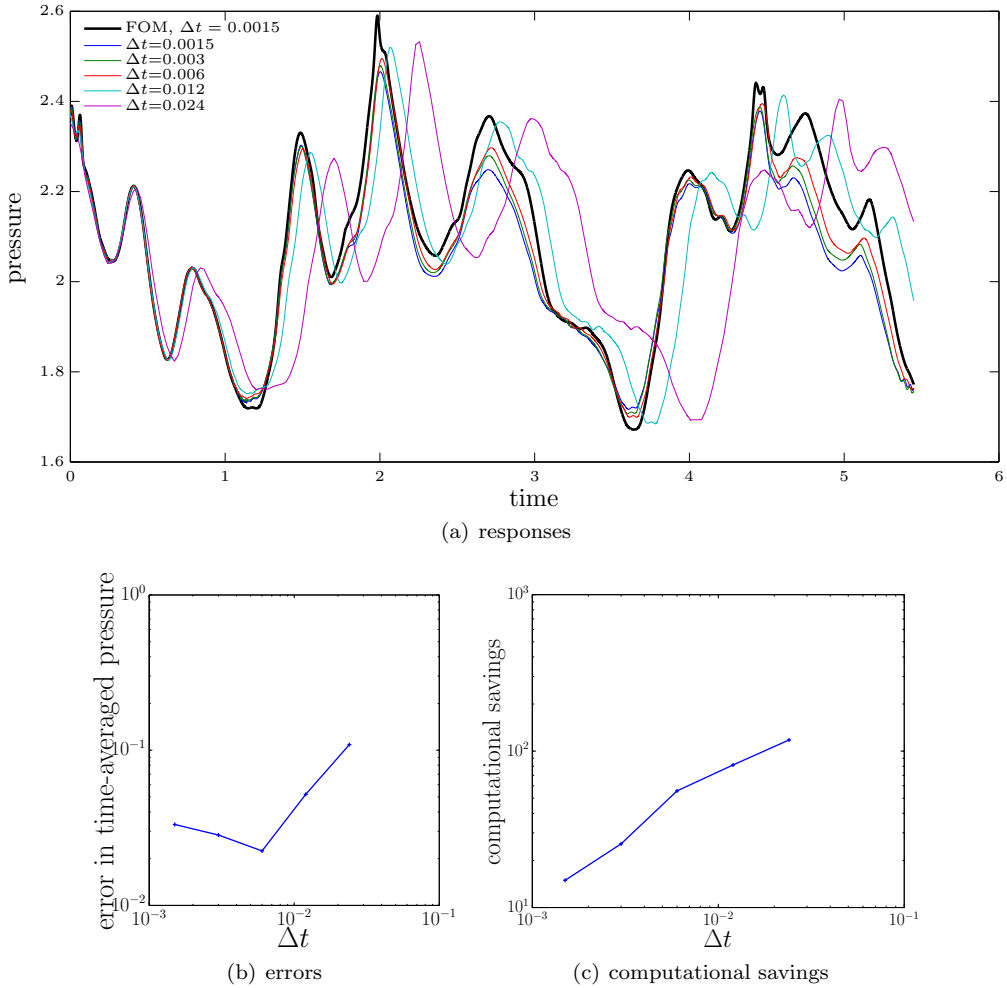
(a) Full domain



(b) Zoom on cavity

Figure 13: Sample mesh (red) embedded within original mesh. The sample mesh is used to to enable low-computational-footprint simulations with the GNAT reduced-order model.

(as measured in core–hours) from 14.9 to 55.7. This highlights that the analysis is also relevant to ROMs equipped with complexity reduction.



(a) responses

(b) errors

(c) computational savings

Figure 14: Responses, errors $\varepsilon(p_{\text{GNAT}}, p_{\text{FOM}_\star})$, and computational savings (as measured in core–hours) produced by the GNAT reduced-order model for different time steps Δt.

7.8. Summary of experimental results

We now briefly summarize the main experimental results:

- Galerkin ROMs are unstable for long time intervals (Figure 7).
- LSPG ROMs are only unstable for small time steps (Figure 7).
- Galerkin and LSPG ROMs are equivalent as $\Delta t \to 0$ (Figure 8).
- LSPG ROMs are more accurate than Galerkin ROMs over small time windows where Galerkin is stable (Figure 9).
- LSPG ROMs are most accurate for an intermediate time step (Figure 9).
- Adding POD modes has the effect of including higher-frequency response components (Figure 10).
- The theoretical error bound for the LSPG ROM exhibits the same time step 'dip' as the experimentally observed error (Figure 11).

- The optimal time step for the LSPG ROM decreases as modes are added to the POD basis (Figure 12).
- Adding modes to the POD basis has little effect on LSPG ROM accuracy for large time steps (Figure 12).
- The optimal time step for the LSPG ROM tends to increase as the time interval increases (Figure 12(g)).
- The GNAT ROM, which is discrete optimal and is equipped with complexity reduction, also produces minimal error for an intermediate time step (Figure 14).

8. Conclusions

This work has performed a comparative theoretical and experimental analysis of Galerkin and LSPG reduced-order models for linear multistep schemes and Runge–Kutta schemes. We have demonstrated a number of new findings that have important practical implications, including conditions under which the LSPG ROM has a time-continuous representation, conditions under which the two techniques are equivalent, and time-discrete error bounds for the two approaches.

Perhaps most surprisingly, we demonstrated that decreasing the time step does not necessarily decrease the error for the LSPG ROM. This phenomenon arose in both the theoretical analysis and in numerical experiments. In particular, our results suggest that the time step should be 'matched' to the spectral content of the reduced basis. In the experiments, we showed that increasing the time step to an intermediate value decreased both the error and the simulation time by an order of magnitude in certain cases. Alternatively, decreasing the time step cause the LSPG ROM to become unstable for longer time intervals. This highlights the critical importance of time-step selection for LSPG ROMs.

Acknowledgments

We thank Prof. Stephen Pope for insightful conversations related to comparing Galerkin and least-squares Petrov–Galerkin reduced-order models; these conversations inspired this work. We also thank the anonymous reviewers for their extremely helpful and insightful comments and suggestions. We also thank Prof. Charbel Farhat for permitting us the use of AERO-F, as well as Julien Cortial, Charbel Bou-Mosleh, and David Amsallem for their previous contributions in implementing nonlinear reduced-order models in AERO-F. K. Carlberg acknowleges an appointment to the Sandia National Laboratories Truman Fellowship in National Security Science and Engineering. The Truman Fellowship is sponsored by Sandia National Laboratories. 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. H. Antil acknowledges the support by the NSF-DMS-1521590. The content of this publication does not necessarily reflect the position or policy of any of these institutions, and no official endorsement should be inferred.

Appendix

Algorithm 1 reports the algorithm for computing a POD basis using normalized snapshots.

References

[1] R. Abgrall and D. Amsallem, *Robust model reduction by l^1-norm minimization and approximation via dictionaries: application to linear and nonlinear hyperbolic problems*, Stanford University Preprint, (2015).

[2] R. Alexander, *Diagonally implicit runge-kutta methods for stiff ODE's*, SIAM Journal on Numerical Analysis, 14 (1977), pp. 1006–1021.

Input: Set of snapshots $\mathcal{X} \equiv \{\boldsymbol{w}_i\}_{i=1}^{n_w} \subset \mathbb{R}^N$, energy criterion $\nu \in [0,1]$
Output: $\boldsymbol{\Phi}(\mathcal{X}, \nu)$

1: Compute thin singular value decomposition $\boldsymbol{W} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T$, where $\boldsymbol{W} \equiv \left[\boldsymbol{w}_1/\|\boldsymbol{w}_1\| \ \cdots \ \boldsymbol{w}_{n_{\boldsymbol{w}}}/\|\boldsymbol{w}_{n_{\boldsymbol{w}}}\|\right]$.

2: Choose dimension of truncated basis $p = n_e(\nu)$, where

$$n_e(\nu) = \arg\min_{i \in \mathcal{V}(\nu)} i$$

$$\mathcal{V}(\nu) := \{n \in \{1, \ldots, n_w\} \mid \sum_{i=1}^{n} \sigma_i^2 / \sum_{i=1}^{n_w} \sigma_i^2 \geq \nu\},$$

and $\boldsymbol{\Sigma} \equiv \operatorname{diag}(\sigma_i)$.

3: $\boldsymbol{\Phi}(\mathcal{X}, \nu) = \left[\boldsymbol{u}^1 \ \cdots \ \boldsymbol{u}^p\right]$, where $\boldsymbol{U} \equiv \left[\boldsymbol{u}^1 \ \cdots \ \boldsymbol{u}^{n_w}\right]$.

Algorithm 1: Proper-orthogonal-decomposition basis computation (normalized snapshots)

[3] D. Amsallem, M. Zahr, and K. Washabaugh, *Fast local reduced basis updates for the efficient reduction of nonlinear systems with hyper-reduction*, Advances in Computational Mathematics, Special Issue on Model Reduction of Parameterized Systems (2015).

[4] S. An, T. Kim, and D. James, *Optimizing cubature for efficient integration of subspace deformations*, ACM Transactions on Graphics (TOG), 27 (2008), p. 165.

[5] H. Antil, S. Field, F. Herrmann, R. Nochetto, and M. Tiglio, *Two-step greedy algorithm for reduced order quadratures*, Journal of Scientific Computing, 57 (2013), pp. 604–637.

[6] H. Antil, M. Heinkenschloss, and D. C. Sorensen, *Application of the discrete empirical interpolation method to reduced order modeling of nonlinear and parametric systems*, vol. 8 of Springer MS&A series: Reduced Order Methods for modeling and computational r G. Rozza, Eds, Springer-Verlag Italia, Milano, 2013.

[7] 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 (2008), pp. 2237–2251.

[8] N. Aubry, P. Holmes, J. L. Lumley, and E. Stone, *The dynamics of coherent structures in the wall region of a turbulent boundary layer*, Journal of Fluid Mechanics, 192 (1988), pp. 115–173.

[9] M. Balajewicz and E. Dowell, *Stabilization of projection-based reduced order models of the navier–stokes equations*, Nonlinear Dynamics, 70 (2012), pp. 1619–1632.

[10] M. Balajewicz, E. Dowell, and B. Noack, *Low-dimensional modelling of high-reynolds-number shear flows incorporating constraints from the navier–stokes equation*, Journal of Fluid Mechanics, 729 (2013), pp. 285–308.

[11] M. F. Barone, I. Kalashnikova, D. J. Segalman, and H. K. Thornquist, *Stable galerkin reduced order models for linearized compressible flow*, Journal of Computational Physics, 228 (2009), pp. 1932–1946.

[12] M. Barrault, Y. Maday, N. C. Nguyen, and A. T. Patera, *An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations*, Comptes Rendus Mathématique Académie des Sciences, 339 (2004), pp. 667–672.

[13] P. Benner, S. Gugercin, and K. Willcox, *A survey of model reduction methods for parametric systems*, SIAM Review, 57 (2015), pp. 483–531.

[14] M. Bergmann, C.-H. Bruneau, and A. Iollo, *Enablers for robust pod models*, Journal of Computational Physics, 228 (2009), pp. 516 – 538.

[15] L. Blackford, A. Cleary, J. Choi, E. d'Azevedo, J. Demmel, I. Dhillon, J. Dongarra, S. Hammarling, G. Henry, A. Petitet, et al., *ScaLAPACK Users' Guide*, Society for Industrial and Applied Mathematics, 1997.

[16] R. Bos, X. Bombois, and P. Van den Hof, *Accelerating large-scale non-linear models for monitoring and control using spatial and temporal correlations*, Proceedings of the American Control Conference, 4 (2004), pp. 3705–3710.

[17] T. Bui-Thanh, K. Willcox, and O. Ghattas, *Model reduction for large-scale systems with high-dimensional parametric input space*, SIAM Journal on Scientific Computing, 30 (2008), pp. 3270–3288.

[18] K. Carlberg, *Model reduction of nonlinear mechanical systems via optimal projection and tensor approximation*, PhD thesis, Stanford University, August 2011.

[19] K. Carlberg, C. Bou-Mosleh, and C. Farhat, *Efficient non-linear model reduction via a least-squares Petrov–Galerkin projection and compressive tensor approximations*, International Journal for Numerical Methods in Engineering, 86 (2011), pp. 155–181.

[20] K. Carlberg, J. Cortial, D. Amsallem, M. Zahr, and C. Farhat, *The GNAT nonlinear model reduction method and its application to fluid dynamics problems*, AIAA Paper 2011-3112, 6th AIAA Theoretical Fluid Mechanics Conference, Honolulu, HI, (2011).

[21] 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*, Journal of Computational Physics, 242 (2013), pp. 623–647.

[22] K. Carlberg, R. Tuminaro, and P. Boggs, *Efficient structure-preserving model reduction for nonlinear mechanical systems with application to structural dynamics*, in AIAA Paper 2012-1969, 53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference, Honolulu, Hawaii, April 23–26 2012.

[23] K. Carlberg, R. Tuminaro, and P. Boggs, *Preserving Lagrangian structure in nonlinear model reduction with application to structural dynamics*, SIAM J. Sci. Comput., 37 (2015), pp. B153—B184.

[24] S. Chaturantabut and D. C. Sorensen, *Nonlinear model reduction via discrete empirical interpolation*, SIAM Journal on Scientific Computing, 32 (2010), pp. 2737–2764.

[25] P. Constantine and Q. Wang, *Residual minimizing model reduction for parameterized nonlinear dynamical systems*, SIAM J. Sci. Comput., 34 (2012), pp. A2118–A2144.

[26] M. Drohmann, B. Haasdonk, and M. Ohlberger, *Reduced basis approximation for nonlinear parametrized evolution equations based on empirical operator interpolation*, SIAM Journal on Scientific Computing, 34 (2012), pp. A937–A969.

[27] R. Everson and L. Sirovich, *Karhunen–Loève procedure for gappy data*, Journal of the Optical Society of America A, 12 (1995), pp. 1657–1664.

[28] F. Fang, C. Pain, I. Navon, A. Elsheikh, J. Du, and D. Xiao, *Non-linear Petrov–Galerkin methods for reduced order hyperbolic equations and discontinuous finite element methods*, Journal of Computational Physics, 234 (2013), pp. 540 – 559.

[29] C. Farhat, P. Avery, T. Chapman, and J. Cortial, *Dimensional reduction of nonlinear finite element dynamic models with finite rotations and energy-based mesh sampling and weighting for computational efficiency*, International Journal for Numerical Methods in Engineering, 98 (2014), pp. 625–662.

[30] C. Farhat, P. Geuzaine, and G. Brown, *Application of a three-field nonlinear fluid-structure formulation to the prediction of the aeroelastic parameters of an F-16 fighter*, Computers & Fluids, 32 (2003), pp. 3–29.

[31] C. Foias, M. Jolly, I. Kevrekidis, and E. Titi, *Dissipativity of numerical schemes*, Nonlinearity, 4 (1991), p. 591.

[32] D. Galbally, K. Fidkowski, K. Willcox, and O. Ghattas, *Non-linear model reduction for uncertainty quantification in large-scale inverse problems*, International Journal for Numerical Methods in Engineering, 81 (2009), pp. 1581–1608.

[33] B. Galletti, C. Bruneau, L. Zannetti, and A. Iollo, *Low-order modelling of laminar flow regimes past a confined square cylinder*, Journal of Fluid Mechanics, 503 (2004), pp. 161–170.

[34] P. Geuzaine, G. Brown, C. Harris, and C. Farhat, *Aeroelastic dynamic analysis of a full F-16 configuration for various flight conditions*, AIAA Journal, 41 (2003), pp. 363–371.

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

[36] A. Iollo, S. Lanteri, and J. A. Desideri, *Stability properties of POD–Galerkin approximations for the compressible Navier–Stokes equations*, Theoretical and Computational Fluid Dynamics, 13 (2000), pp. 377–396.

[37] M. Jolly, I. Kevrekidis, and E. Titi, *Preserving dissipation in approximate inertial forms for the kuramoto-sivashinsky equation*, Journal of Dynamics and Differential Equations, 3 (1991), pp. 179–197.

[38] I. Kalashnikova and M. Barone, *On the stability and convergence of a galerkin reduced order model (rom) of compressible flow with solid wall and far-field boundary treatment*, International journal for numerical methods in engineering, 83 (2010), pp. 1345–1375.

[39] S. Lall, P. Krysl, and J. Marsden, *Structure-preserving model reduction for mechanical systems*, Physica D: Nonlinear Phenomena, 184 (2003), pp. 304–318.

[40] P. A. LeGresley, *Application of proper orthogonal decomposition (POD) to design decomposition methods*, PhD thesis, Stanford University, 2006.

[41] M. Marion and R. Temam, *Nonlinear Galerkin methods*, SIAM Journal on Numerical Analysis, 26 (1989), pp. 1139–1157.

[42] N. Ngoc Cuong, K. Veroy, and A. T. Patera, *Certified real-time solution of parametrized partial differential equations*, in Handbook of Materials Modeling, S. Yip, ed., Springer Netherlands, 2005, pp. 1529–1564.

[43] B. R. Noack, P. Papas, and P. A. Monkewitz, *The need for a pressure-term representation in empirical galerkin models of incompressible shear flows*, Journal of Fluid Mechanics, 523 (2005), pp. 339–365.

[44] C. Prud'homme, D. V. Rovas, K. Veroy, L. Machiels, Y. Maday, A. T. Patera, and G. Turinici, *Reliable real-time solution of parameterized partial differential equations: Reduced-basis output bound methods*, Journal of Fluids Engineering, 124 (2002), pp. 70–80.

[45] M. Rathinam and L. R. Petzold, *A new look at proper orthogonal decomposition*, SIAM Journal on Numerical Analysis, 41 (2003), pp. 1893–1925.

[46] D. Rempfer, *On low-dimensional galerkin models for fluid flow*, Theoretical and Computational Fluid Dynamics, 14 (2000), pp. 75–88.

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

[48] G. Rozza, D. B. P. Huynh, and A. T. Patera, *Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations*, Archives of Computational Methods in Engineering, 15 (2008), pp. 229–275.

[49] D. Ryckelynck, *A priori hyperreduction method: an adaptive approach*, Journal of Computational Physics, 202 (2005), pp. 346–366.

[50] O. San and T. Iliescu, *Proper orthogonal decomposition closure models for fluid flows: Burgers equation*, arXiv preprint arXiv:1308.3276, (2013).

[51] J. Shen, *Long time stability and convergence for fully discrete nonlinear galerkin methods*, Applicable Analysis, 38 (1990), pp. 201–229.

[52] S. Sirisup and G. Karniadakis, *A spectral viscosity method for correcting the long-term behavior of pod models*, Journal of Computational Physics, 194 (2004), pp. 92–116.

[53] L. Sirovich, *Turbulence and the dynamics of coherent structures. III: dynamics and scaling*, Quarterly of Applied Mathematics, 45 (1987), pp. 583–590.

[54] P. Spalart and S. Allmaras, *A one-equation turbulence model for aerodynamic flows*, La Recherche Aérospatiale, 1 (1994), p. 5.

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

[56] K. Veroy, C. Prud'homme, D. V. Rovas, and A. T. Patera, *A posteriori error bounds for reduced-basis approximation of parametrized noncoercive and nonlinear elliptic partial differential equations*, AIAA Paper 2003-3847, 16th AIAA Computational Fluid Dynamics Conference, Orlando, FL, (2003).

[57] 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 (2012), pp. 10–26.