Masayuki Yano

6 Model reduction in computational aerodynamics

Abstract: Computational aerodynamics has become an indispensable tool in the design and analysis of modern aircraft. However, traditional high-fidelity aerodynamics simulations can be computationally too expensive for scenarios that require responses in real time (e. g., flow control) and/or predictions for many different configurations (e. g., design-space exploration and flight-parameter sweep). The goal of model reduction is to accelerate the solution of unsteady and/or parameterized aerodynamics problems in real-time and/or many-query scenarios. In this chapter, we survey model reduction techniques for linearized and nonlinear aerodynamics problems that have been developed in the past two decades. We discuss essential ingredients of model reduction: stable and efficient projection methods, generation of the reduced basis tailored for the specific solution manifold, and offline-online computational decomposition. We focus on techniques that are designed to address challenges in aerodynamics – nonlinearity, limited stability, limited regularity, and wide range of scales – and have been demonstrated for multidimensional aerodynamic flows. We highlight successful applications of model reduction for large-scale aerodynamics problems.

Keywords: aerodynamics, model reduction, parameterized partial differential equations, (Petrov–)Galerkin projection, reduced basis

MSC 2010: 65N30, 65N35, 35Q30, 35Q35, 76G25

6.1 Introduction

6.1.1 Motivation

With advances in both computational algorithms and hardware, computational fluid dynamics (CFD) has become an indispensable tool in the analysis and design of aerospace vehicles. Today's CFD tools can accurately predict aerodynamics of aircraft in cruise conditions and complement wind-tunnel and flight tests in the aircraft design process; in fact, with the advances in CFD, the number of wings tested in the design of a typical commercial aircraft has decreased by an order of magnitude from the late 1970s to the early 2000s [38].

However, there are computational challenges that still remain out of reach for traditional CFD solvers. To motivate the model reduction work reviewed in this chapter, we name a few "grand challenges" outlined in vision papers [51, 61]. First is high-

Masayuki Yano, University of Toronto, Toronto, Ontario, Canada

https://doi.org/10.1515/9783110499001-006

fidelity aerodynamic database generation; the task requires accurate prediction of aerodynamic forces for the entire range of flight conditions with variations in, e. g., the free stream Mach number and angle of attack. Second is real-time dynamic flight simulation; the task requires aerodynamic or aeroelastic simulation of maneuvering aircraft with the control input specified in real-time. Third is probabilistic design of cooled turbine blades; the task requires accurate characterization of the turbine blade performance under geometric uncertainties due to manufacturing variabilities. These tasks are challenging for traditional CFD solvers because they require (i) predictions for a large number of configurations (i. e., many-query) and/or (ii) real-time predictions of transient phenomena. Completing these tasks, especially in the time scale and computational resources available in typical engineering settings, can be prohibitive with traditional CFD tools. The objective of this chapter is to survey the state of the art in model reduction for many-query and/or real-time problems in aerodynamics.

6.1.2 Real-time and many-query scenarios

We now provide examples of many-query and/or real-time engineering scenarios to which model reduction has been applied. We restrict ourselves to problems in aerodynamics, rather than more general fluid dynamics; we refer to Chapter 9 of this volume for the latter. We do not attempt to provide a comprehensive review; we merely present a few representative works.

S1. *Aerodynamic shape optimization*. One of the many-query applications of model reduction in aerodynamics is shape optimization. Reduced-order models (ROMs) are used to accelerate aerodynamics analysis under parametric geometry changes and to optimize the geometry. The task consists of three steps: parameterization of the geometry; construction of a ROM; and identification of the optimal geometry. ROMs have been used in many-query analysis [6, 69] and inverse design, where the objective is to identify airfoil geometry that yields the prescribed pressure distribution [43, 44, 45, 78].

S2. *Flight-parameter sweep*. Another many-query application of parametric model reduction in aerodynamics is flight-parameter sweep. ROMs are used to accelerate the prediction of aerodynamic forces and moments for a range of flight conditions described in terms of the angle of attack and Mach number [80, 79, 66, 68, 75, 76].

S3. *Aeroelasticity*. One of the classical real-time applications of model reduction in aerodynamics is aeroelasticity. The goal is to analyze the interaction between aerodynamics forces and elastic structure and to detect, for instance, the onset of flutter. Aeroelasticity saw one of the earliest uses of model reduction, with works appearing in at least as early as the mid-1990s for nonparameterized problems [33, 55, 42, 34, 64]. More recently, techniques have been extended to parameterized aeroelasticity problems, with the angle of attack and Mach number as parameters [47, 46, 4, 2, 5]. We also note that there are nonprojection-based approaches to model reduction, e. g., by the Volterra series; however, given the focus of this handbook, we do not cover these works and refer interested readers to review papers [26, 49, 25].

S4. *Model predictive control.* Another real-time application of model reduction is the control of aerodynamic systems using model predictive control (MPC). Without model reduction, MPC is infeasible for large-scale systems, as it requires real-time solution of optimization problems. ROMs have been incorporated in MPC to control shock location in a supersonic diffuser [36] and to optimize flight path under fuel consumption and aeroelastic constraints [3].

S5. *Uncertainty quantification and state estimation.* Model reduction has also been used for uncertainty quantification, in which the effect of geometry or flow-condition uncertainties are propagated to quantities of interests. ROMs have been used for probabilistic analysis of turbine blades, in which simulation is carried out for thousands of different configurations [17]. Model reduction has also been applied to state estimation, where the aerodynamic flow field is inferred from surface pressure tap data [16, 71].

6.1.3 Scope and outline

We make four disclaimers regarding the scope of this chapter. First, we restrict our presentation to works on aerodynamics rather than more general fluid mechanics, and in particular to works on compressible flow rather than incompressible flow. We refer to Chapter 9 of this volume for more general coverage of model reduction in CFD. Second, given the emphasis of this handbook, we focus on formulation, rather than theoretical, aspects of model reduction. We however note that mathematical theories have played important roles in the development of model reduction approaches for aerodynamics problems; we refer to references provided throughout the chapter for further theoretical discussions. Third, the model reduction literature for aerodynamics problems is vast, with development from both engineering and applied mathematics communities; we attempt to cover representative works but admit the coverage is not exhaustive and there are inevitable omissions. Fourth, we note that (i) precise requirements for an ROM depend on the particular engineering scenario and there is no universal formulation suitable for all scenarios; (ii) even for a given scenario there are many different approaches; and (iii) there are relatively few comparative studies due to the recentness of some of the techniques and the shear cost of performing such studies for large-scale aerodynamics problems. We hence do not attempt to make definitive recommendations and focus on surveying existing approaches, with a hope that the chapter will still serve as a guide to construct an ROM that works for the problem of interest.

This chapter is organized as follows. In Section 6.2, we review full-order discretizations for aerodynamics problems. In Section 6.3, we review model reduction techniques for linearized aerodynamics problems; the linearized problem is relevant for small perturbation analysis, which arises in applications including aeroelasticity, flow control, and uncertainty quantification. In Section 6.4, we review model reduction techniques for nonlinear aerodynamics equations; the full nonlinear analysis is often required for aerodynamic shape optimization and flight-parameter sweep.

6.2 Full-order models

In this section we review full-order models (FOMs) for aerodynamics problems. We consider both the linearized and full nonlinear FOMs; the associated ROMs will be constructed in Sections 6.3 and 6.4, respectively. We describe FOMs in abstract forms to accommodate various governing equations and discretizations under a unified framework.

6.2.1 Conservation laws of aerodynamics

We introduce the general form of aerodynamics partial differential equations (PDEs) considered throughout this chapter. We introduce a P-dimensional parameter domain $\mathcal{P} \in \mathbb{R}^P$, a d-dimensional spatial domain $\Omega \subset \mathbb{R}^d$, the associated boundary $\partial\Omega$, and a time interval $\mathcal{I} \equiv (0, T] \subset \mathbb{R}$. Aerodynamic flow in Ω over $\mathcal{I}$ is modeled by a system of N_c nonlinear conservation laws of the form

$$\begin{aligned} \frac{\partial u}{\partial t} + \nabla \cdot \big(f^{\mathrm{inv}}(u) + f^{\mathrm{visc}}(u, \nabla u)\big) &= f^{\mathrm{src}}(u, \nabla u) \quad \text{in } \Omega \times \mathcal{I}, \\ b\big(u, n \cdot f^{\mathrm{visc}}(u, \nabla u)\big) &= 0 \quad \text{on } \partial\Omega \times \mathcal{I}, \\ u|_{t=0} &= u^0 \quad \text{in } \Omega, \end{aligned} \tag{6.1}$$

where u is the conservative state, f^{inv} is the inviscid flux function, f^{visc} is the viscous flux function, f^{src} is the source function, b is the boundary condition function, and u^0 is the initial state. While the exact forms of the flux, source, and boundary functions depend on the specific governing equation – the Euler, Navier–Stokes, or Reynolds-averaged Navier–Stokes (RANS) equations – and flow conditions, all conservation laws in aerodynamics can be cast in the general form (6.1). We also emphasize that, although omitted here for brevity, all functions in general depend on the parameter $\boldsymbol{\mu} \in \mathcal{P}$ for parameterized problems and the time $t \in \mathcal{I}$ for unsteady problems.

In many aerodynamics problems, our interest is not necessarily in the entire state field u but in few quantities of interest (i. e., output). Arguably the most common output in aerodynamics are lift and drag, which can be expressed as a surface integral of the form

$$s \equiv \int_{\Gamma_{\mathrm{body}}} f^{\mathrm{out}}(u, n \cdot f^{\mathrm{visc}}(u, \nabla u); n) ds,$$

where $\Gamma_{\mathrm{body}} \subset \partial\Omega$ is the aerodynamic surface of interest, n denotes the unit vector normal to Γ_{body}, and the function f^{out} maps the surface state and viscous flux to aerodynamic forces.

We make a few remarks about the governing equations in aerodynamics. First, inviscid flows are modeled by the Euler equations, which are purely hyperbolic. Second, viscous flows are modeled by the Navier–Stokes equations which, for Reynolds number relevant to aerodynamics, are convection-dominated. Third, for turbulent flow simulations based on the RANS equations, the Navier–Stokes equations are augmented with additional empirical PDEs that model the turbulence behavior; most turbulence models are highly nonlinear, including the one-equation Spalart–Allmaras (SA) turbulence model [62] used in most of the works reviewed in this chapter. Fourth, nonconservative variables, such as the entropy variables [35], may be used as the working state variables; the entropy variables are of particular interest for stability analysis of Galerkin methods [12] and in particular ROMs [9, 39].

6.2.2 Semi-discrete form

We now consider a full-order approximation of the conservation law (6.1). While there is a number of different discretizations for (6.1), they must provide stability for hyperbolic and convection-dominated PDEs. As a result, most works on model reduction for aerodynamics use one the three full-order discretizations: a finite volume method [65], a stabilized finite element method [15, 37], or a discontinuous Galerkin (DG) method [24, 7]. We refer to the references above for details of the discretizations, and here describe FOMs in an abstract form.

To introduce an FOM, we first introduce a triangulation $\mathcal{T}_h \equiv \{\kappa_1, \ldots, \kappa_{N_e}\}$, where $\{\kappa_i\}_{i=1}^{N_e}$ is a set of N_e nonoverlapping elements such that $\overline{\Omega} = \cup_{\kappa \in \mathcal{T}_h} \overline{\kappa}$ and $\kappa_i \cap \kappa_j = \emptyset$, $i \neq j$. We next introduce an N_h-dimensional approximation space $V_h \subset V$ associated with $\mathcal{T}_h$; the associated dual space is denoted by V_h' with the duality pairing $\langle \cdot, \cdot \rangle : V_h' \times V \to \mathbb{R}$. We then introduce an FOM spatial residual operator $r_h : V_h \times \mathcal{P} \to V_h'$; the particular form of the residual depends on the conservation laws and discretization. A semi-discrete form of our FOM problem is as follows: Given $\boldsymbol{\mu} \in \mathcal{P}$, find $u_h(t; \boldsymbol{\mu}) \in V_h$, $t \in \mathcal{I}$, such that

$$\frac{\partial u_h(t; \boldsymbol{\mu})}{\partial t} + r_h(u_h(t; \boldsymbol{\mu}); \boldsymbol{\mu}) = 0 \quad \text{in } V_h', \tag{6.2}$$

and $u_h(t = 0; \boldsymbol{\mu}) = \Pi_h u^0(\boldsymbol{\mu})$; here $u^0(\boldsymbol{\mu}) \in V$ is the initial condition, and $\Pi_h : V \to V_h$ is a projection operator from V to V_h. Throughout this chapter, for any Hilbert space W

and the associated dual space W', the statement "$g = 0$ in W'" should be interpreted as $\langle g, w\rangle = 0\ \forall w \in W$. We then introduce an FOM output functional $q_h : V_h \times \mathcal{P} \to \mathbb{R}^{N_o}$, so that the set of N_o outputs is given by

$$s_h(t;\boldsymbol{\mu}) = q_h(u_h(t;\boldsymbol{\mu});\boldsymbol{\mu}). \tag{6.3}$$

We assume that the solution and output to the FOM exists and is unique.

We may also consider an "algebraic form" of the problem, i. e., the form of the problem described by matrices and vectors, which is convenient for the computational implementation of the formulation. To this end, we first introduce a basis $\{\varphi^j\}_{j=1}^{N_h}$ of the space V_h. We next associate any function $v_h \in V_h$ with a generalized coordinate $\mathbf{v}_h \in \mathbb{R}^{N_h}$ by $v_h = \mathbf{v}_h^j \varphi^j$, where $\mathbf{v}_h^j$ denotes the j-th component of $\mathbf{v}_h$ and the summation on the repeated indices is implied. We then introduce algebraic forms of the FOM residual operator $\mathbf{r}_h : \mathbb{R}^{N_h} \times \mathcal{P} \to \mathbb{R}^{N_h}$, the output functional $\mathbf{q}_h : \mathbb{R}^{N_h} \times \mathcal{P} \to \mathbb{R}$, and the mass matrix $\mathbf{M}_h \in \mathbb{R}^{N_h \times N_h}$ given by

$$\begin{aligned}
\mathbf{r}_h(\mathbf{w}_h;\boldsymbol{\mu})_i &\equiv \langle r_h(\mathbf{w}_h^j\varphi^j;\boldsymbol{\mu}), \varphi^i\rangle, \quad i = 1,\dots,N_h,\\
\mathbf{q}_h(\mathbf{w}_h;\boldsymbol{\mu}) &\equiv q_h(\mathbf{w}_h^j\varphi^j;\boldsymbol{\mu}),\\
\mathbf{M}_{h,ij} &\equiv (\varphi^j,\varphi^i)_{L^2(\Omega)}, \quad i,j = 1,\dots,N_h.
\end{aligned}$$

The algebraic form of the FOM problem (6.2) and (6.3) is as follows: Given $\boldsymbol{\mu} \in \mathcal{P}$, find $\mathbf{u}_h(t;\boldsymbol{\mu}) \in \mathbb{R}^{N_h}$, $t \in \mathcal{I}$, such that

$$\mathbf{M}_h \frac{d\mathbf{u}_h(t;\boldsymbol{\mu})}{dt} + \mathbf{r}_h(\mathbf{u}_h(t;\boldsymbol{\mu});\boldsymbol{\mu}) = 0 \quad \text{in } \mathbb{R}^{N_h} \tag{6.4}$$

for $\mathbf{u}_h(t = 0;\boldsymbol{\mu}) = \mathbf{u}_h^0(\boldsymbol{\mu})$, and then evaluate

$$s_h(t;\boldsymbol{\mu}) = \mathbf{q}_h(\mathbf{u}_h(t;\boldsymbol{\mu});\boldsymbol{\mu}). \tag{6.5}$$

This algebraic form of the problem is equivalent to the operator form (6.2) and (6.3); in particular, $u_h(t;\boldsymbol{\mu}) = \mathbf{u}_h^j(t;\boldsymbol{\mu})\varphi^j$. The solution to (6.4) is typically obtained using a Newton-like method.

We make a few remarks. First, for a typical aerodynamics problem, $P = \mathcal{O}(1-10)$, $N_h = \mathcal{O}(10^5 - 10^7)$, and $N_o = \mathcal{O}(1)$. Second, for steady problems, the time derivative term vanishes and we seek $u(\boldsymbol{\mu}) \in V_h$ such that

$$r_h(u(\boldsymbol{\mu});\boldsymbol{\mu}) = 0 \quad \text{in } V_h', \tag{6.6}$$

or, equivalently, $\mathbf{u}_h(\boldsymbol{\mu}) \in \mathbb{R}^{N_h}$ such that $\mathbf{r}_h(\mathbf{u}_h(\boldsymbol{\mu});\boldsymbol{\mu}) = 0$ in $\mathbb{R}^{N_h}$. Third, for problems with shape deformations, the spatial domain Ω depends on the parameter $\boldsymbol{\mu} \in \mathcal{D}$; we refer to a review [57] for the treatment of parameter-dependent domains by a reference-domain formulation, which provides an equivalent problem in a parameter-independent reference domain. Fourth, while finite volume methods are typically not presented as a weak formulation (6.2), the form encompasses (in general high-order) finite volume methods, as the methods can be recast as a DG method with an appropriate state reconstruction function; see, e. g., [13]. Fifth, in any event, all FOM discretizations can be expressed in the algebraic form (6.4) and (6.5). Hence, in Sections 6.3 and 6.4, we describe all model reduction techniques using this abstract framework.

6.2.3 Full-discrete form

We now introduce a full-discrete form of the FOM (6.4). We first introduce time instances $0 = t^0 \leq t^1 \leq \cdots \leq t^K = T$, and the associated sequence of functions $\{u_h^k(\boldsymbol{\mu})\}_{k=1}^K = \{\mathbf{u}_h^{k,j}(\boldsymbol{\mu})\varphi^j\}_{k=1}^K$ such that $u_h(t^k;\boldsymbol{\mu}) \approx u_h^k(\boldsymbol{\mu})$, $k = 1,\ldots,K$. We then discretize the semi-discrete equation (6.4) using a multistep or multistage scheme. For instance, if the backward Euler method is used, the full-discrete FOM problem is as follows: Given $\boldsymbol{\mu} \in \mathcal{P}$, find $\{\mathbf{u}_h^k(\boldsymbol{\mu}) \in \mathbb{R}^{N_h}\}_{k=1}^K$ such that

$$\mathbf{r}_{h,\Delta t}^k(\mathbf{u}_h^k(\boldsymbol{\mu}); \mathbf{u}_h^{k-1}(\boldsymbol{\mu}); \boldsymbol{\mu}) \equiv \frac{1}{\Delta t}\mathbf{M}_h(\mathbf{u}_h^k(\boldsymbol{\mu}) - \mathbf{u}_h^{k-1}(\boldsymbol{\mu})) + \mathbf{r}_h(\mathbf{u}_h^k(\boldsymbol{\mu}); \boldsymbol{\mu}) = 0$$

for $k = 1,\ldots,K$, and $\mathbf{u}_h^{k=0}(\boldsymbol{\mu}) = \mathbf{u}^0(\boldsymbol{\mu})$. Here, $\mathbf{r}_{h,\Delta t}^k : \mathbb{R}^{N_h} \times \mathbb{R}^{N_h} \times \mathcal{P} \to \mathbb{R}^{N_h}$ is the full-discrete residual operator for the backward Euler method at the time instance k, which depends on the state at the previous time step $\mathbf{u}_h^{k-1}(\boldsymbol{\mu})$. More generally, for a multistep method, the full-discrete residual operator depends on the states at k_{step} previous time instances and takes the form $\mathbf{r}_{h,\Delta t}^k : \mathbb{R}^{N_h} \times \mathbb{R}^{N_h \times k_{\text{step}}} \times \mathcal{P} \to \mathbb{R}^{N_h}$. We assume that an appropriate time marching scheme is chosen such that a sequence of stable solutions exists.

We note that the solution to the steady problem (6.6) is often obtained using a pseudo-transient continuation (PTC) method [41], which solves the unsteady problem using pseudo-time stepping, to improve the convergence of the nonlinear solver. Hence the temporal stability is an important consideration even for steady problems. We refer to [41] for a review of PTC methods.

6.2.4 Linearized equations

While aerodynamic flow is governed by a system of nonlinear conservation laws, as discussed in Section 6.1, time-dependent linearized analysis is also of engineering interest. The goal of linearized analysis is to propagate small input disturbances to output perturbations. Here, the input disturbances may result from small changes in the geometry (e. g., vibrations), boundary conditions (e. g., gust), or initial conditions; our interest is in the associated change in the aerodynamic forces and moments.

Before we proceed, we make one notational change. In the previous section, we introduced the parameter-dependent steady residual operator $r_h : V_h \times \mathcal{P} \to V_h'$; in this section, to be consistent with literature on linearized aerodynamics analysis, we explicitly separate the parameters subjected to input disturbances from those that are not. Specifically, we introduce a Q-dimensional input space $\mathcal{Q} \subset \mathbb{R}^Q$. We then introduce the steady residual operator $r_h : V_h \times \mathcal{Q} \times \mathcal{P} \to V_h'$, which is a function of the state, input, and parameter. Similarly, we introduce the output operator $q_h : V_h \times \mathcal{Q} \times \mathcal{P} \to \mathbb{R}^{N_o}$.

In linearized analysis, we decompose the solution $u_h \in V_h$ into a base solution $\bar{u}_h$ and perturbation δu_h so that $u_h = \bar{u}_h + \delta u_h$. Similarly, we decompose the input $v \in \mathcal{Q}$ into a base input $\bar{v}$ and disturbance δv so that $v = \bar{v} + \delta v$. The perturbation is governed by the following linearized problem: Given $\boldsymbol{\mu} \in \mathcal{P}$ and input $\delta v(t) \in \mathcal{Q}$, find $\delta u_h(t; \delta v, \boldsymbol{\mu}) \in V_h$, $t \in \mathcal{I}$, such that

$$\frac{\partial \delta u_h(t; \delta v, \boldsymbol{\mu})}{\partial t} + J_h(\boldsymbol{\mu}) \delta u_h(t; \delta v, \boldsymbol{\mu}) + B_h(\boldsymbol{\mu}) \delta v(t) = 0 \quad \text{in } V_h', \tag{6.7}$$

and $\delta u_h(t = 0; \boldsymbol{\mu}) = \Pi_h \delta u^0(\boldsymbol{\mu})$ for $\delta u^0(\boldsymbol{\mu}) \in V_h$ the initial perturbation. Here, the Jacobian $J_h(\boldsymbol{\mu}) \in \mathcal{L}(V_h, V_h')$ is the Fréchet derivative of $r_h(\cdot, \bar{v}; \boldsymbol{\mu}) : V_h \to V_h'$ at $\bar{u}_h$, and the operator $B_h(\boldsymbol{\mu}) \in \mathcal{L}(Q, V_h')$ is the Fréchet derivative of $r_h(\bar{u}_h, \cdot; \boldsymbol{\mu}) : \mathcal{Q} \to V_h'$ at $\bar{v}$. Given the perturbed state $\delta u_h(t; \delta v, \boldsymbol{\mu}) \in V_h$, we evaluate the associated output perturbation

$$\delta s_h(t; \delta v, \boldsymbol{\mu}) = g_h(\boldsymbol{\mu}) \delta u_h(t; \delta v, \boldsymbol{\mu}),$$

where $g_h(\boldsymbol{\mu}) \in \mathcal{L}(V_h, \mathbb{R}^{N_o})$ is the Fréchet derivative of $q_h(\cdot, \bar{v}; \boldsymbol{\mu})$ at $\bar{u}_h$. The goal of the linearized aerodynamics analysis is to map the disturbances in the input $\delta v(t) \in \mathcal{Q}$ to the perturbations in the output $\delta s_h(t; \delta v, \boldsymbol{\mu}) \in \mathbb{R}^{N_o}$ for any parameter value $\boldsymbol{\mu} \in \mathcal{P}$. In aerodynamics, the linearization state $\bar{u} \in \mathcal{V}_h$ is often the solution to the steady-state nonlinear problem (6.6); i. e., $r_h(\bar{u}_h; \boldsymbol{\mu}) = 0$ in V_h'.

The linearized equations can also be expressed in an algebraic form. To this end, we introduce the Jacobian matrix $\mathbf{J}_h(\boldsymbol{\mu}) \in \mathbb{R}^{N_h \times N_h}$, input matrix $\mathbf{B}_h(\boldsymbol{\mu}) \in \mathbb{R}^{N_h \times Q}$, and output gradient vector $\mathbf{g}_h(\boldsymbol{\mu}) \in \mathbb{R}^{N_o \times N_h}$ such that

$$\begin{aligned}
\mathbf{J}_h(\boldsymbol{\mu})_{ij} &= \langle J_h(\bar{u}_h; \boldsymbol{\mu}) \varphi^j, \varphi^i \rangle, \quad i, j = 1, \ldots, N_h, \\
\mathbf{B}_h(\boldsymbol{\mu})_{ij} &= \langle B_h(\bar{u}_h; \boldsymbol{\mu}) e^j, \varphi^i \rangle, \quad i = 1, \ldots, N_h,\ j = 1, \ldots, Q, \\
\mathbf{g}_h(\boldsymbol{\mu})_{ij} &= \langle g_h(\bar{u}_h; \boldsymbol{\mu}) \varphi^j, e^i \rangle \quad i = 1, \ldots, N_o,\ j = 1, \ldots, N_h,
\end{aligned}$$

where e^j is the unit vector with the j-th entry equal to 1. The algebraic form of the linearized problem is as follows: Given $\boldsymbol{\mu} \in \mathcal{P}$ and input $\delta v(t) \in \mathcal{Q}$, find $\delta \mathbf{u}_h(t; \delta v, \boldsymbol{\mu}) \in \mathbb{R}^{N_h}$, $t \in \mathcal{I}$, such that

$$\mathbf{M}_h \frac{d \delta \mathbf{u}_h(t; \delta v, \boldsymbol{\mu})}{dt} + \mathbf{J}_h(\boldsymbol{\mu}) \delta \mathbf{u}_h(t; \delta v, \boldsymbol{\mu}) + \mathbf{B}_h(\boldsymbol{\mu}) \delta v(t) = 0 \quad \text{in } \mathbb{R}^{N_h}, \tag{6.8}$$

and evaluate the output

$$\delta s_h(t; \delta v, \boldsymbol{\mu}) = \mathbf{g}_h(\boldsymbol{\mu}) \delta \mathbf{u}_h(t; \delta v, \boldsymbol{\mu}).$$

We note that, for a fixed parameter $\boldsymbol{\mu} \in \mathcal{P}$, the problem is in the standard linear time-invariant form. The application of a time marching scheme yields a full-discrete form of the linearized equations whose solution $\{\delta \mathbf{u}_h^k\}_{k=1}^K$ satisfies $\delta \mathbf{u}_h^k \approx \delta \mathbf{u}_h(t^k)$, $k = 1, \ldots, K$, analogously to the discussion for the nonlinear FOM in Section 6.2.3.

6.3 Model reduction for linearized aerodynamics

In this section we discuss model reduction of linearized aerodynamics problems. As discussed in Section 6.2.4, linearized (i. e., small-perturbation) analysis of unsteady aerodynamics provides significant insights in many engineering scenarios. More pragmatically, model reduction of linearized PDEs requires fewer ingredients than that of nonlinear PDEs, and hence we introduce common ingredients in the linearized context.

6.3.1 Galerkin method

We now consider reduced-order approximations of the FOM (6.7) (or equivalently (6.8)). To this end, we introduce a sequence of reduced basis spaces $V_{N=1} \subset \cdots \subset V_{N=N_{\max}}$, each of which is a subset of $\mathcal{V}_h$; for a typical aerodynamics ROM, $N_{\max} = \mathcal{O}(10\text{–}100)$, which is significantly smaller than $N_h = \mathcal{O}(10^5\text{–}10^7)$. We then introduce the associated hierarchical reduced basis $\{\zeta^n \in \mathcal{V}_h\}_{n=1}^N$ such that $V_N = \operatorname{span}\{\zeta^n\}_{n=1}^N$, $N = 1, \ldots, N_{\max}$. We may also express the reduced basis in an algebraic form $\{\boldsymbol{\zeta}^n \in \mathbb{R}^{N_h}\}_{n=1}^N$ such that $\zeta^n = \boldsymbol{\zeta}^{n,j} \varphi^j$, $N = 1, \ldots, N_{\max}$; we introduce the associated reduced basis matrix $\mathbf{Z}_N = (\boldsymbol{\zeta}^1, \ldots, \boldsymbol{\zeta}^N) \in \mathbb{R}^{N_h \times N}$. We will discuss various methods to construct the reduced basis in Section 6.3.2; for now we assume the basis is given.

Given a reduced basis space V_N, the semi-discrete form of the Galerkin-ROM problem is as follows: Given $\boldsymbol{\mu} \in \mathcal{P}$ and $\delta v(t) \in \mathcal{Q}$, find $\delta u_N(t; \delta v, \boldsymbol{\mu}) \in V_N$, $t \in \mathcal{I}$, such that

$$\frac{\partial \delta u_h(t; \delta v, \boldsymbol{\mu})}{\partial t} + J_h(\boldsymbol{\mu}) \delta u_N(t; \delta v, \boldsymbol{\mu}) + B_h(\boldsymbol{\mu}) \delta v(t) = 0 \quad \text{in } V_N', \tag{6.9}$$

and $\delta u_N(t = 0; \boldsymbol{\mu}) = \Pi_N u^0(\boldsymbol{\mu})$, where $\Pi_N : V \to V_N$ is a projection operator from V to V_N. Again, for $g \in V_h'$, the statement $g = 0$ in V_N' should be interpreted as $\langle g, v \rangle = 0$ $\forall v \in V_N$. We then evaluate the output perturbation $\delta s_h(t; \delta v, \boldsymbol{\mu}) = g_h(\boldsymbol{\mu}) \delta u_N(t; \delta v, \boldsymbol{\mu})$. The comparison of the FOM problem (6.7) and the Galerkin-ROM problem (6.9) shows that the latter results from the restriction of the test and trail spaces to the reduced space $V_N \subset V_h$.

The Galerkin-ROM problem (6.9) can also be expressed in an algebraic (or matrix-vector) form. To this end, we associate any function $v_N \in V_N$ with a generalized coordinate $\mathbf{v}_N \in \mathbb{R}^N$ by $v_N = \mathbf{v}_N^j \zeta^j$; we may also express the full-order generalized coordinate of $v_N \in V_N$ as $\mathbf{v}_h = \mathbf{v}_N^j \boldsymbol{\zeta}^j = \mathbf{Z}_N \mathbf{v}_N \in \mathbb{R}^{N_h}$. Given the reduced basis, we define the ROM operators

$$\begin{aligned}
\mathbf{M}_N &\equiv \mathbf{Z}_N^T \mathbf{M}_h \mathbf{Z}_N = \big((\zeta^j, \zeta^i)_{L^2(\Omega)}\big)_{i,j=1}^N \in \mathbb{R}^{N\times N},\\
\mathbf{J}_N(\boldsymbol{\mu}) &\equiv \mathbf{Z}_N^T \mathbf{J}_h(\boldsymbol{\mu}) \mathbf{Z}_N = \big(\langle J_h(\boldsymbol{\mu})\zeta^j, \zeta^i\rangle\big)_{i,j=1}^N \in \mathbb{R}^{N\times N},\\
\mathbf{B}_N(\boldsymbol{\mu}) &\equiv \mathbf{Z}_N^T \mathbf{J}_h(\boldsymbol{\mu}) = \big(\langle B_h(\boldsymbol{\mu}) e^j, \zeta^i\rangle\big)_{i=1,j=1}^{N,Q} \in \mathbb{R}^{N\times Q},\\
\mathbf{g}_N(\boldsymbol{\mu}) &\equiv \mathbf{g}_h(\boldsymbol{\mu}) \mathbf{Z}_N = \big(\langle g_h(\boldsymbol{\mu})\zeta^j, e^i\rangle\big)_{j=1}^N \in \mathbb{R}^{N_o\times N}.
\end{aligned} \tag{6.10}$$

The algebraic form of the linearized problem is as follows: Given $\boldsymbol{\mu} \in \mathcal{P}$ and $\delta v(t;\boldsymbol{\mu}) \in \mathcal{Q}$, find $\delta \mathbf{u}_N(t;\delta v, \boldsymbol{\mu}) \in \mathbb{R}^N$, $t \in \mathcal{I}$, such that

$$\mathbf{M}_N \frac{d\delta \mathbf{u}_N(t;\delta v,\boldsymbol{\mu})}{dt} + \mathbf{J}_N(\boldsymbol{\mu})\delta \mathbf{u}_N(t;\boldsymbol{\mu}) + \mathbf{B}_N(\boldsymbol{\mu})\delta v = 0 \quad \text{in } \mathbb{R}^N, \tag{6.11}$$

and evaluate the output $\delta s_N(t;\delta v,\boldsymbol{\mu}) = \mathbf{g}_N(\mathbf{u}_N(\boldsymbol{\mu});\delta v,\boldsymbol{\mu})\delta \mathbf{u}_N(t;\boldsymbol{\mu})$. Again, the operator form (6.9) and the algebraic form (6.11) are equivalent and $\delta u_N(t;\boldsymbol{\mu}) = \delta \mathbf{u}_N^j(t;\boldsymbol{\mu})\zeta^j$. We note that the ROM operators (6.10) are precomputed in the construction stage, so that the ROM (6.11) can be solved in $\mathcal{O}(N^\bullet)$ operations for the exponent $\bullet$ between 1 and 3. In particular the cost to solve the ROM (6.11) is independent of N_h; we recall that $N = \mathcal{O}(10\text{–}100)$ and $N_h = \mathcal{O}(10^5\text{–}10^7)$ for a typical aerodynamics problem. We discuss this offline-online computational decomposition in Section 6.3.4.

6.3.2 Reduced basis for nonparameterized linearized problems

The efficacy of the Galerkin-ROM (6.9) (or (6.11)) depends on the choice of the reduced basis. We now review techniques to identify an effective reduced basis $\{\zeta^j\}_{j=1}^N$ (or reduced basis matrix $\mathbf{Z}_N \in \mathbb{R}^{N_h\times N}$). For practical and historical reasons, we first present procedures for nonparameterized (or fixed-parameter) problems; the model reduction of time-varying but fixed-parameter aerodynamics problems enables fast simulation of complex flows, which is essential for, for instance, MPC. As the problems are nonparameterized, we suppress the argument $\boldsymbol{\mu}$ for all operators throughout this section. In addition, as our primary goal is to provide recipes for implementation, rather than to discuss theory, we present algorithms in algebraic forms.

6.3.2.1 Eigenmodes

A classical approach to identify a reduced basis for linearized aerodynamics problems is eigenanalysis. The approach, first introduced by Hall [33], is as follows:

1. Solve the generalized eigenproblem: Find the eigenvector $\boldsymbol{\zeta}^k \in \mathbb{R}^{N_h}$ and the associated eigenvalue $\lambda^k \in \mathbb{C}$ such that

$$\mathbf{J}_h \boldsymbol{\zeta}^k = \lambda^k \mathbf{M}_h \boldsymbol{\zeta}^k \quad \text{in } \mathbb{R}^{N_h};$$

   without loss of generality, sort the eigenpairs such that $|\lambda^1| \geq \cdots \geq |\lambda^{N_h}|$.
2. Construct the reduced basis matrix $\mathbf{Z}_N = (\boldsymbol{\zeta}^1, \ldots, \boldsymbol{\zeta}^N) \in \mathbb{R}^{N_h \times N}$.

While historically important, eigenanalysis has a major limitation: The reduced basis is based solely on the Jacobian $\mathbf{J}_h$ and does not account for the system input $\mathbf{B}_h$ or output $\mathbf{g}_h$. Hence, the number of eigenmodes N required to achieve a given solution or output accuracy is typically greater than empirical approaches based on proper orthogonal decomposition (POD).

6.3.2.2 Time-domain POD

To address limitations of eigenmodes discussed in Section 6.3.2.1, Romanowski [55] proposes a (time-domain) POD approach for linearized Euler equations. We here present the method of snapshots [60] to efficiently compute a POD basis for large-scale problems in aerodynamics:

1. Choose L time-dependent training inputs $\{\{\delta v^l(t)\}_{t\in\mathcal{I}}\}_{l=1}^L$, where l is the training input index.
2. Solve the full-discrete form of the linearized FOM (6.7) for the training inputs $\{\delta v^l\}_{l=1}^L$ and for K time steps $\{t^k\}_{k=1}^K$ to construct a snapshot matrix $\mathbf{S} \in \mathbb{R}^{N_h \times N_s}$, whose columns are $\delta \mathbf{u}_h^k(\delta v^l) \approx \delta \mathbf{u}_h(t^k; \delta v^l)$ for $k = 1, \ldots, K$, $l = 1, \ldots, L$, and $N_s \equiv KL$.
3. Construct the correlation matrix $\mathbf{A} = \mathbf{S}^T \mathbf{X}_h \mathbf{S}$ in $\mathbb{R}^{N_s \times N_s}$. Here, $\mathbf{X}_h \in \mathbb{R}^{N_h \times N_h}$ such that $\mathbf{X}_{h,ij} = (\varphi^j, \varphi^i)_{X_h}$ is associated with an appropriate inner product; a common choice is the $L^2(\Omega)$-inner product.
4. Solve the eigenproblem: Find $(\mathbf{v}^k, \lambda^k) \in \mathbb{R}^{N_s} \times \mathbb{R}$ such that

$$\mathbf{A}\mathbf{v}^k = \lambda^k \mathbf{v}^k \quad \text{in } \mathbb{R}^{N_s};$$

   without loss of generality, sort the eigenpairs such that $|\lambda^1| \geq \cdots \geq |\lambda^{N_s}|$.
5. Set the reduced basis matrix $\mathbf{Z}_N = (\boldsymbol{\zeta}^1, \ldots, \boldsymbol{\zeta}^N) \in \mathbb{R}^{N_h \times N}$, where

$$\boldsymbol{\zeta}^k = \lambda_k^{-1/2} \mathbf{S} \mathbf{v}^k, \quad k = 1, \ldots, N.$$

The resulting basis $\mathbf{Z}_N \in \mathbb{R}^{N_h \times N}$ is orthogonal with respect to the $\mathbf{X}_h$ inner product; i. e., $\mathbf{Z}_N^T \mathbf{X}_h \mathbf{Z}_N = I_N$. In addition, $\mathbf{Z}_N$ minimizes the X_h-projection error for the snapshots; i. e., $\mathbf{Z}_N = \arg\min_{\mathbf{W}_N \in \mathbb{R}^{N_h \times N}} \|\mathbf{S} - \mathbf{W}_N \mathbf{W}_N^T \mathbf{X}_h \mathbf{S}\|_{\mathbf{X}_h}$. In this sense, the POD basis is optimal for the approximation of the state $\delta u_h(t; \delta v)$ associated with the particular system input δv; however, the system output s_h is not accounted for in the POD method.

6.3.2.3 Frequency-domain POD

A variant of the time-domain POD approach above is the frequency-domain POD approach proposed by Kim [42] and Hall et al. [34]. As the name suggests, this approach takes advantage of the linearity of the problem (6.8) and computes snapshots in the frequency domain. Namely, we consider time-harmonic disturbances of the form $\delta v(t) = \delta\hat{v}e^{j\omega t}$ of a frequency $\omega \in \mathbb{R}$ so that the associated time-harmonic perturbations are of the form $\delta\mathbf{u}_h(t;\delta v,\boldsymbol{\mu}) = \delta\hat{\mathbf{u}}_h(\delta\hat{v})e^{j\omega t}$ for $\delta\hat{\mathbf{u}}_h(\delta\hat{v}) = (j\omega\mathbf{M}_h + \mathbf{J}_h)^{-1}\mathbf{B}_h\delta\hat{v}$, where $j \equiv \sqrt{-1}$. The frequency-domain POD approach replaces the first two steps of the time-domain POD approach in Section 6.3.2.2 with the following:

1'. Choose L training inputs $\{\delta\hat{v}^l \in \mathbb{R}^Q\}_{l=1}^L$ and K training frequencies $\{\omega_k \in \mathbb{R}\}_{k=1}^K$.
2'. Solve the frequency-domain equation

$$(j\omega_k\mathbf{M}_h + \mathbf{J}_h)\delta\hat{\mathbf{u}}^k(\delta\hat{v}^l) = \mathbf{B}_h\delta\hat{v}^l \tag{6.12}$$

for $\{\delta\hat{v}^l\}_{l=1}^L$ and $\{\omega_k\}_{k=1}^K$ to construct a snapshot matrix $\mathbf{S} \in \mathbb{R}^{N_h\times N_s}$, whose columns are the real and imaginary parts of the frequency-domain perturbation, $\mathbb{R}(\delta\hat{\mathbf{u}}^k(\delta v^l))$ and $\mathbb{J}(\delta\hat{\mathbf{u}}^k(\delta v^l))$, for $k = 1,\dots,K$, $l = 1,\dots,L$, and $N_s \equiv 2KL$.

The training input modes and frequencies can be chosen based on known characteristics of input disturbances; e. g., for aeroelasticity problems, the modes and frequencies may be chosen to coincide with the resonance modes of the structure. For linearized aerodynamics problems, frequency-domain POD is often more efficient than time-domain POD and hence is preferred; the approach has been successfully applied to the linearized Euler equations in works including [42, 34, 64, 47, 46, 4, 2]. We however make two cautionary remarks: First, implementation must support complex arithmetic; second, just like time-domain POD, while the POD basis is in some sense optimized for the solution field $\delta u_h(t;\delta v) \in V_h$, it is not specialized for the particular system output s_h.

6.3.2.4 Balanced POD

The time- and frequency-domain POD approaches construct a reduced space V_N which is well suited for the approximation of the entire state $\delta u_h(t;\delta v) \in V_h$; however, in aerodynamics, we are often not interested in the entire state but rather only in few outputs (i. e., quantities of interest). In these cases, we can construct a more efficient ROM using the balanced POD (BPOD) method proposed by Willcox and Peraire [73], which approximates balanced truncation [52] for large-scale problems. The key to BPOD is (i) to realize that both the input and output play equally important roles in characterizing the input-output relationship and (ii) to incorporate the dual problem to account for the choice of the output. The dual problem for the linearized aerodynamics problem (6.8) with a single output ($N_o = 1$) is as follows: Given $\delta v(t) \in \mathcal{Q}$, find

$\mathbf{z}_h(t;\delta\nu) \in \mathbb{R}^{N_h}$, $t \in \mathcal{I}$, such that

$$-\frac{d\mathbf{z}_h(t;\delta\nu)}{dt} + \mathbf{J}_h^T\mathbf{z}(t;\delta\nu) + \mathbf{g}^T = 0 \quad \text{in } \mathbb{R}^{N_h}, \tag{6.13}$$

and then evaluate the output

$$\delta s_h(t;\delta\nu) = \delta\nu\mathbf{B}_h^T\mathbf{z}_h(t;\delta\nu).$$

The associated frequency-domain problem seeks $\delta\hat{\mathbf{z}}(\delta\nu) = (-j\omega\mathbf{M}_h+\mathbf{J}_h^T)^{-1}\mathbf{g}_h^T$. The BPOD procedure based on frequency-domain sampling for $N_o = 1$ is as follows:

1. Choose L training inputs $\{\delta\hat{\nu}^l \in \mathbb{R}^Q\}_{l=1}^L$ and K training frequencies $\{\omega_k \in \mathbb{R}\}_{k=1}^K$.
2. Solve the frequency-domain problem (6.12) to collect N_s primal snapshots, and then obtain the POD mode matrix $\mathbf{Z}_p^{\text{pr}} \in \mathbb{R}^{N_h\times p}$ and eigenvalue matrix $\mathbf{\Lambda}_p^{\text{pr}} \in \mathbb{R}^{p\times p}$ for the $p \geq N$ largest eigenvalues.
3. Solve the frequency-domain dual problem (6.13) to collect N_s adjoint snapshots, and then obtain the POD mode matrix $\mathbf{Z}_p^{\text{du}} \in \mathbb{R}^{N_h\times p}$ and eigenvalue matrix $\mathbf{\Lambda}_p^{\text{du}} \in \mathbb{R}^{p\times p}$ for the p largest eigenvalues.
4. Compute the eigenvectors $\mathbf{Z}_N \in \mathbb{R}^{N_h\times N}$ associated with the N largest eigenvalues of the matrix $(\mathbf{Z}_p^{\text{pr}}\mathbf{\Lambda}_p^{\text{pr}}\mathbf{Z}_p^{\text{pr}\,T})(\mathbf{Z}_p^{\text{du}}\mathbf{\Lambda}_p^{\text{du}}\mathbf{Z}_p^{\text{du}\,T})$ using a Krylov subspace method. (Note that the matrix is never explicitly formed.)

The BPOD method produces a reduced basis optimized for the input-output mapping problem and enables goal-oriented reduction of linearized aerodynamics problems [73]; depending on the output, BPOD significantly reduces the dimension of the reduced space required to achieve a given output tolerance compared to the standard POD, as demonstrated for a two-dimensional plunging airfoil [73]. A variant of BPOD modified for a problem with a large number of outputs is developed by Rowley in [56].

6.3.2.5 Other goal-oriented methods

We survey a few other goal-oriented methods to generate reduced bases; we again restrict ourselves to techniques that have been demonstrated for aerodynamics problems. In [74], Willcox et al. propose an Arnoldi-based method, which identifies a reduced basis by matching moments of the FOM input-output transfer function, and apply it to aeroelastic analysis of a transonic turbine cascade with unsteady blade motions. In [72], Willcox and Megretski propose a method which identifies a reduced basis by computing the Fourier expansion of the discrete-frequency transfer function, and apply it to analysis of a supersonic diffuser. In [18], Bui-Thanh et al. propose a more general approach to goal-oriented model reduction that identifies a reduced basis as a solution of a constrained optimization problem and apply it to analysis of a subsonic turbine blade. All three methods are goal-oriented in the sense that they consider both system inputs and outputs to identify an effective reduced basis.

6.3.3 Reduced basis for parameterized linearized problems

We have so far discussed the construction of reduced bases for nonparameterized problems or, equivalently, for one fixed parameter. For parameterized problems, in general a reduced basis constructed for one parameter value does not provide a good approximation for another parameter value, as the associated dynamics can be very different; see, for example, a study for parameterized turbine blades by Epureanu [28]. We here discuss a few different strategies to construct reduced bases for parameterized problems.

6.3.3.1 Global POD

One approach to construct a reduced basis for parameterized problems is to prepare a "global" or "composite" POD basis, which has been trained for a range of parameters, as proposed for aerodynamics problems by Schmit, Taylor, and Glauser [59, 63]. In this approach, we first introduce a training parameter set $\Xi_{N_\mathrm{t}} \equiv \{\boldsymbol{\mu}^m\}_{m=1}^{N_\mathrm{t}}$, collect the snapshots for all parameter values, and then apply POD to the snapshots. The global POD approach for parameterized problem replaces the first two steps of the time-domain POD approach in Section 6.3.2.2 with the following:

1'. Choose N_t training parameters $\{\boldsymbol{\mu}^n\}_{n=1}^{N_\mathrm{t}}$ and L training inputs $\{\delta v\}_{l=1}^{L}$.
2'. Solve the full-discrete form of the linearized FOM (6.7) for the training parameters $\{\boldsymbol{\mu}^m\}_{m=1}^{N_\mathrm{t}}$, training inputs $\{\delta v^l\}_{l=1}^{L}$, and time steps $\{t^k\}_{k=1}^{K}$ to construct a snapshot matrix $\mathbf{S} \in \mathbb{R}^{N_h \times N_\mathrm{s}}$, whose columns are $\delta \mathbf{u}_h^k(\delta v^l; \boldsymbol{\mu}^m) \approx \delta \mathbf{u}_h(t^k; \delta v^l)$ for $k = 1, \ldots, K$, $l = 1, \ldots, L$, $m = 1, \ldots, N_\mathrm{t}$, and $N_\mathrm{s} \equiv KLN_\mathrm{t}$.

The global POD approximation works well for problems with a relatively small parameter dimension and extent; however, the method may suffer from two issues if the problem exhibits significant parametric variations. First, the FOM may need to be solved for a large number of training parameters, which results in a high training cost. Second, a large number of POD modes may be required to accurately capture the dynamics. (More precisely, if the Kolmogorov N-width of the parametric manifold $\{u_h(t; \delta v, \boldsymbol{\mu})\}_{t \in \mathcal{I}, \delta v \in \mathcal{Q}, \boldsymbol{\mu} \in \mathcal{P}}$ is large, then a large number of modes is required to achieve sufficient accuracy.)

6.3.3.2 The (weak) greedy algorithm

To address the potentially high training cost associated with the global POD, the (weak) greedy algorithm has been developed [67, 57]. The greedy algorithm successively identifies a reduced basis $\{\boldsymbol{\zeta}^j\}_{j=1}^{N}$ based on the behavior of a rapidly computable error estimate $\eta_N(\boldsymbol{\mu})$. The algorithm takes as the input the training parameter set $\Xi_t \subset \mathcal{D}$ which reasonably covers the domain. Then, in the N-th iteration, given $\mathbf{Z}_{N-1} \in \mathbb{R}^{N_h \times (N-1)}$ the algorithm proceeds as follows:

1. Find the parameter with the largest error estimate: $\boldsymbol{\mu}^N = \arg\max_{\boldsymbol{\mu} \in \Xi_t} \eta_{N-1}(\boldsymbol{\mu})$.
2. Solve the FOM for $\boldsymbol{\mu}^N$ to obtain $\mathbf{u}_h(\boldsymbol{\mu}^N) \in \mathbb{R}^{N_h}$.
3. Augment the reduced basis with the new snapshot: $\mathbf{Z}_N = (\mathbf{Z}_{N-1}, \mathbf{u}_h(\boldsymbol{\mu}^N))$; re-orthonormalize $\mathbf{Z}_N$ using Gram–Schmidt.

The steps are repeated until the user-prescribed error tolerance is met for all $\boldsymbol{\mu} \in \Xi_t$. For unsteady problems, Step 3 incorporates an additional reduction technique (e. g., POD) to compress the multiple temporal snapshots associated with a single unsteady solve; this approach, called POD-greedy algorithm, was proposed and analyzed in [32] and its variant is applied to probabilistic analysis of turbine cascades in [17].

The weak greedy algorithm has two advantages over POD. First, it requires only N FOM solutions compared to $N_t \gg N$ solutions for global POD; hence it reduces the training cost, and a larger Ξ_t can be used for more exhaustive training. Second, in the presence of a goal-oriented error estimate, the ROM trained will meet the error threshold for the engineering quantities of interest at least for $\boldsymbol{\mu} \in \Xi_t$. However, one major limitation of the weak greedy algorithm is that it requires a rapidly computable error estimate; due to the difficulty of constructing such an error estimate for hyperbolic and convection-dominated problems in aerodynamics, the greedy algorithm has seen somewhat limited use in the field. In addition, while the training cost is reduced relative to global POD, the resulting ROM may still require a large N if the problem exhibits significant parametric variations. We refer to a review paper [57] for more detailed description of the weak greedy algorithm.

6.3.3.3 Parameter-domain decomposition

One approach to reduce the ROM size for problems that exhibit a large parameter extent is to decompose the parameter domain $\mathcal{P}$ (or time interval $\mathcal{I}$) into smaller subdomains to limit the parameter extent, which in turn controls the reducibility (i. e., the Kolmogorov N-width) of the parametric manifold. Namely, we first subdivide $\mathcal{P}$ into $N_{\mathcal{P}}$ subdomains $\{\mathcal{P}^n\}_{n=1}^{N_{\mathcal{P}}}$ so that $\bigcup_{n=1}^{N_{\mathcal{P}}} \overline{\mathcal{P}}^n = \overline{\mathcal{P}}$. We then construct a set of $N_{\mathcal{P}}$ reduced bases $\{\mathbf{Z}^n\}_{n=1}^{N_{\mathcal{P}}}$ for the parametric manifolds $\{\{\mathbf{u}_h(\boldsymbol{\mu})\}_{\boldsymbol{\mu} \in \mathcal{P}^n}\}_{n=1}^{N_{\mathcal{P}}}$. To make an ROM prediction for a given parameter $\boldsymbol{\mu} \in \mathcal{P}$, we identify the subdomain $\mathcal{P}^n$ such that $\boldsymbol{\mu} \in \mathcal{P}^n$ and then invoke the ROM.

One of the earliest applications of the parameter-domain decomposition approach in aerodynamics is Annonen et al. [6]; the so-called multi-POD approach considers multiple reduced bases associated with different shape deformations. Washabaugh et al. [68] also employ the approach for Mach number sweep of a full aircraft configuration. Some versions of the reduced space interpolation methods [2], which is discussed in Section 6.3.3.4, also incorporates the idea to work with a database of reduced spaces. We also refer to [27] for detailed analyses of parameter-domain decomposition approaches.

6.3.3.4 Reduced-space interpolation based on Grassmann manifold

Another approach to reduce the ROM size for problems that exhibit a large parameter extent is to "interpolate" a set of reduced spaces computed for several parameter values to construct a new reduced space for the particular parameter value. One simple idea is to interpolate each basis vector $\boldsymbol{\zeta}^j$ as a function of $\boldsymbol{\mu} \in \mathcal{P}$; however, this approach, which works with the vectors and not the space, is shown to work poorly for aeroelasticity problems [48]. To address the problem, Lieu et al. [48, 47, 46] propose the so-called subspace-angle interpolation method to interpolate any two reduced spaces. Subsequently, Amsallem et al. [4, 2] propose a more general approach to interpolate an arbitrary number of reduced spaces associated with $\{\mathbf{Z}_N^i\}_{i=1}^{N_Z}$ to construct a new reduced basis $\mathbf{Z}_N$. The approach builds on the observation that the reduced space V_N spanned by a reduced basis $\mathbf{Z}_N$ is an element of the Grassmann manifold $G(N, N_h)$. To interpolate reduced spaces, the approach (i) invokes a logarithmic map to map reduced spaces onto a tangent space, (ii) performs standard interpolation in the tangent space, and (iii) invokes an exponential map to map back the logarithmic representation of the interpolated basis to identify $\mathbf{Z}_N$. Here we outline the algorithm:

1. Choose parameter values $\{\boldsymbol{\mu}^i\}_{i=0}^{N_Z}$ and construct the associated reduced bases $\{\mathbf{Z}_N^i\}_{i=0}^{N_Z}$; $i = 0$ is the reference point.
2. Compute the logarithms $\{\boldsymbol{\Gamma}_i \in \mathbb{R}^{N_h \times N}\}_{i=1}^{N_Z}$ given by
$$\begin{aligned}(\mathbf{I} - \mathbf{Z}_N^0 \mathbf{Z}_N^{0\,T})\mathbf{Z}_N^i(\mathbf{Z}_N^{0\,T}\mathbf{Z}_N^i)^{-1} &= \mathbf{U}_i \boldsymbol{\Sigma}_i \mathbf{V}_i \quad \text{in } \mathbb{R}^{N_h \times N},\\ \boldsymbol{\Gamma}_i &= \mathbf{U}_i \tan^{-1}(\boldsymbol{\Sigma}_i)\mathbf{V}_i^T \quad \text{in } \mathbb{R}^{N_h \times N},\end{aligned}$$
where the right-hand side of the first step is the thin singular value decomposition of the matrix in the left-hand side.
3. Given $\boldsymbol{\mu} \in \mathcal{P}$, interpolate each entry of the parameter-logarithm-matrix pairs $(\boldsymbol{\mu}^i, \boldsymbol{\Gamma}^i)_{i=1}^{N_Z}$ using a multivariate interpolation scheme for $\mathbb{R}^P$ to find $\boldsymbol{\Gamma} \in \mathbb{R}^{N_h \times N}$ associated with $\boldsymbol{\mu} \in \mathcal{P}$.
4. Compute the exponential map of the logarithm $\boldsymbol{\Gamma} \in \mathbb{R}^{N_h \times N}$ given by
$$\begin{aligned}\boldsymbol{\Gamma} &= \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T,\\ \mathbf{Z}_N &= \mathbf{Z}_N^0 \mathbf{V}\cos(\boldsymbol{\Sigma}) + \mathbf{U}\sin(\boldsymbol{\Sigma}).\end{aligned}$$

This interpolation method on the Grassmann manifold can be thought of as a generalization of the subspace-angle interpolation method [48, 47, 46]; the two methods are equivalent when $N_Z = 2$ reduced bases are used for interpolation, but the former generalizes to an arbitrary number of reduced bases [4]. For problems with a large parameter extent, the reduced space interpolation methods can also be combined with the parameter-domain decomposition method discussed in Section 6.3.3.3; in this case, the interpolation is performed on a subset of all available reduced bases [2]. The reduced basis interpolation methods have been demonstrated for parameterized aeroelastic analysis of full aircraft configurations [4, 2] as discussed further in Section 6.3.6.

6.3.4 Offline-online computational decomposition

As briefly discussed in Section 6.3.1, model reduction achieves computational speedup through offline-online computational decomposition. The offline stage is expensive but is performed only once. The online stage is cheap, and it is invoked in real-time for many different inputs and/or parameters. To describe offline-online computational decomposition for linearized aerodynamics problems, we break down the model reduction procedure into three steps:

1. Collect the FOM snapshots and construct a reduced basis $\mathbf{Z}_N$ (or reduced bases $\{\mathbf{Z}_N^n\}$) using a method described in Section 6.3.2 or 6.3.3.
2. Construct the ROM operators by projecting the FOM operators onto the reduced basis $\mathbf{Z}_N$ according to (6.10).
3. Given the input $\delta v(t) \in \mathcal{Q}$, $t \in \mathcal{I}$, solve the ROM problem (6.11).

In general, Step 1 is the most expensive stage, as it requires time- or frequency-domain solutions of the FOM for a number of different control inputs and/or parameters. Step 2 also requires access to the FOM, and hence does require $\mathcal{O}(N_h)$ operations; however, this step is much cheaper than Step 1, as performing the projection (6.10) is much cheaper than solving the FOM (6.11). Step 3, which works exclusively with the ROM, requires $\mathcal{O}(N^\bullet)$ operations; since $N_h = \mathcal{O}(10^5 - 10^7)$ and $N = \mathcal{O}(10 - 100)$ for a typical aerodynamics problem, the ROM achieves significant computational reduction relative to the FOM.

The offline-online computational decomposition takes on different forms depending on whether the problem is parameterized. For nonparameterized problems, the offline stage comprises Steps 1 and 2; first a reduced basis is identified using a method in Section 6.3.2, and then the ROM is constructed in terms of the reduced operators (6.10). In the online stage, given an input $\delta v(t) \in \mathcal{Q}$, $t \in \mathcal{I}$, we invoke the ROM (6.11); note that the online stage requires only $\mathcal{O}(N^\bullet)$ operations.

For parameterized problems, the offline stage comprises only Step 1; either a global reduced basis or a set of reduced bases is constructed using a method in Section 6.3.3. In the online stage, given $\boldsymbol{\mu} \in \mathcal{P}$, we first identify an appropriate reduced basis: For the parameter-domain decomposition method discussed in Section 6.3.3.3, this step requires the identification of the subdomain $\mathcal{P}^n$ to which $\boldsymbol{\mu}$ belongs; for the reduced space interpolation method discussed in Section 6.3.3.4, this step involves the interpolation of the reduced bases. We then perform Step 2; project the FOM operators onto $\mathbf{Z}_N$ to identify the ROM operators (6.10). We finally invoke the ROM to approximate the linearized aerodynamics problem for the given $\boldsymbol{\mu} \in \mathcal{P}$ and $\delta v(t) \in \mathcal{Q}$, $t \in \mathcal{I}$. Unlike the online stage for nonparameterized problems, the online stage for parameterized problems requires the access to the FOM in Step 2 and hence requires $\mathcal{O}(N_h)$ operations. Nevertheless, significant speedup can be achieved relative to the FOM as Step 2 is still much cheaper than the unsteady solution of the FOM.

We note that *if* the parameterized FOM operators admit a decomposition that is affine in functions of parameters, then the associated reduced operators can be precomputed in the offline stage and hence the online cost would be $\mathcal{O}(N^\bullet)$; however, most of the relevant problems in aerodynamics do not admit this so-called affine parameter decomposition. We refer to a review paper [57] for offline-online computational decomposition in the presence of affine parameter decomposition. We also note that it may be appropriate to invoke the empirical interpolation method [10, 31] or its variant to identify an approximate affine decomposition; see Chapter 5 of Volume 2.

6.3.5 Stability of the Galerkin-ROM

As discussed in Section 6.1, the focus of this handbook is on formulation and not theory. However, as time stability of ROMs (6.7) is one of the key issues in model reduction of linearized aerodynamics problems, we briefly mention relevant literature; here, time stability refers to the ability to bound some norm of the solution $\|u(t)\|_\star$ by the initial state and boundary conditions.

Barone et al. [9] and Kalashnikova et al. [39] analyze the time stability of the Galerkin-ROM (6.9). The works show that the ROM is stable if the symmetrized form of the hyperbolic system is used with appropriate boundary conditions. We note that for compressible Euler and Navier–Stokes equations (i) the symmetrized system is described in the entropy variables [35, 11]; (ii) the associated energy norm is given by $(w, v)_{A_0} = \int_\Omega v^T A_0 w dx$, where A_0 is the Jacobian of the conservative variables with respect to the entropy variables; and (iii) the mass matrix in (6.7) is also modified accordingly. Kalashnikova et al. [40] further extend the stability analysis to aeroelasticity problems where the structure is modeled by a linearized von Kármán plate equation.

In addition to analysis, we note there are ROM formulations that are designed to achieve guaranteed stability; we again restrict ourselves to works that have been demonstrated for aerodynamics problems. The Fourier-based formulation of Willcox and Megretski [72] discussed in Section 6.3.2.5, for instance, is guaranteed to preserve stability of the underlying FOM; the method has been applied to model reduction [72] and MPC [36] for which POD yields unstable ROMs. Amsallem and Farhat [5] also propose an online-efficient stabilization based on Petrov–Galerkin projection and apply it to aeroelastic analysis of a wing-store configuration.

6.3.6 Large-scale applications

We conclude this section on model reduction for linearized aerodynamics problems with a few applications to large-, industry-scale problems.

- *Aeroelastic analysis of the AGARD model 445.6 wing* [64]. In this work Thomas et al. consider flutter prediction of a weakened AGARD model 445.6 wing. The FOMs consist of $N_h \approx 2.6 \times 10^5$ to 7.8×10^5 aerodynamic degrees of freedom. The flutter boundaries for six different values of the base flow Mach number are analyzed. For each flight Mach number, the snapshots are computed for the first five structural resonance modes ($\{\delta\hat{v}^l\}$) and six frequencies ($\{\omega^k\}$); POD is applied to identify a ROM with $N = 55$ modes. The ROM is then used to construct the root loci with respect to the reduced velocity and to provide accurate predictions of flutter velocities.
- *Aeroelastic analysis of a full F-16 aeroelastic configuration* [2]. In this work Amsallem et al. consider model reduction of a full aeroelastic F-16 configuration. The FOM consists of $N_h \approx 2 \times 10^6 + 1.7 \times 10^5$ aerodynamic and structural degrees of freedom, respectively. The parameters are the base flow Mach number and angle of attack. In the offline stage, a set of reduced bases for 83 different flight configurations are prepared using the frequency-domain POD approach; each basis comprises $N = 90$ modes. In the online stage, the reduced bases are interpolated on a manifold as discussed in Section 6.3.3.4. For the five predictive test configurations considered, the error in the $L^2(\mathcal{I})$-norm of the unsteady lift varies from 0.4 % to 7 %. The time to solve the linearized system is reduced by a factor of 90 in the online stage. (However, the online stage also requires the computational of the steady-state equilibrium solution; when this step is taken into account, the overall speedup factor is approximately 7.) The aeroelasticity problem is also considered in [47, 46, 4].
- *Probabilistic analysis of unsteady turbine blades* [17]. In this work Bui-Thanh et al. consider model reduction of a two-blade turbine system to analyze the effect of geometric uncertainties on unsteady lift forces. The FOM consists $N_h \approx 1 \times 10^5$ degrees of freedom. The geometric modes are identified using principal component analysis on data from 145 real blades; geometric perturbations are parameterized using $P = 10$ parameters. The reduced basis are identified using a greedy algorithm modified for the high-dimensional parameter space; the resulting ROM consists of $N = 290$ modes. The reduced model is then invoked for 10,000 different geometries to estimate the distribution of the work per cycle (WPC). Relative to the FOM, the ROM achieves less than 0.5 % error in the mean and 2 % error in

the variance. The time to complete the 10,000 analyses is reduced from 516 hours for the FOM to 1.1 hours for the ROM, a computational reduction by a factor of 468.

6.4 Model reduction for nonlinear aerodynamics

In this section we discuss model reduction of nonlinear aerodynamics problems. While linearized analysis suffices for some aerodynamics scenarios, applications such as shape optimization and flight-parameter sweep require full nonlinear analysis. As some of the model reduction ingredients are the same as those discussed for linearized problems in Section 6.3, we focus on techniques and challenges that are unique to full nonlinear analysis.

6.4.1 Projection methods

While the Galerkin method is by far the most common approach for model reduction of linearized aerodynamics problems, there are a few different projection methods that are commonly used for nonlinear aerodynamics problems. We here review the two most popular methods, the Galerkin and minimum-residual methods, and provide a short discussion of other approaches.

6.4.1.1 Galerkin method

We first introduce the Galerkin approximation of the nonlinear aerodynamics problem (6.2). As in Section 6.3.1, we assume that a sequence of reduced basis spaces $V_{N=1} \subset \cdots \subset V_{N=N_{\max}}$ and the associated hierarchical reduced basis $\{\zeta^j\}_{j=1}^N$ is given; we discuss the procedures to generate the reduced basis in Section 6.4.3. The semi-discrete form of the Galerkin-ROM problem is as follows: Given $\boldsymbol{\mu} \in \mathcal{P}$, find $u_N(t;\boldsymbol{\mu}) \in V_N$, $t \in \mathcal{I}$, such that

$$\frac{\partial u_N(t;\boldsymbol{\mu})}{\partial t} + r_h(u_N(t;\boldsymbol{\mu});\boldsymbol{\mu}) = 0 \quad \text{in } V_N', \tag{6.14}$$

and $u_N(t=0;\boldsymbol{\mu}) = \Pi_N u^0(\boldsymbol{\mu})$. Again, for $g \in V_h'$, the statement $g = 0$ in V_N' should be interpreted as $\langle g, v\rangle = 0\ \forall v \in V_N$. We then evaluate the output $s_N(t;\mu) = q_h(u_N(t;\mu);\mu)$.

We may also consider the algebraic form of the problem. We recall from Section 6.3.1 that we associate any function $v_N \in V_N$ with a generalized coordinate $\mathbf{v}_N \in \mathbb{R}^N$ by $v_N = \mathbf{v}_N^j \zeta^j$; we may also express the FOM generalized coordinate of $v_N \in V_N$ as $\mathbf{v}_h = \mathbf{v}_N^j \boldsymbol{\zeta}^j = \mathbf{Z}_N \mathbf{v}_N \in \mathbb{R}^{N_h}$. Given the basis, we define the ROM residual $\mathbf{r}_N : \mathbb{R}^N \times \mathcal{P} \to \mathbb{R}^N$, output functional $\mathbf{q}_N : \mathbb{R}^N \times \mathcal{P} \to \mathbb{R}$, and mass matrix $\mathbf{M}_N \in \mathbb{R}^{N\times N}$

such that

$$\begin{aligned}
\mathbf{r}_N(\mathbf{w}_N;\boldsymbol{\mu}) &\equiv \mathbf{Z}_N^T \mathbf{r}_h(\mathbf{Z}_N\mathbf{w}_N;\boldsymbol{\mu}) = (\langle r_h(\mathbf{w}_N^j\zeta^j;\boldsymbol{\mu}),\zeta^i\rangle)_{i=1}^N,\\
\mathbf{q}_N(\mathbf{w}_N;\boldsymbol{\mu}) &\equiv \mathbf{q}_h(\mathbf{Z}_N\mathbf{w}_N;\boldsymbol{\mu}) = q_h(\mathbf{w}_N^j\zeta^j;\boldsymbol{\mu}),\\
\mathbf{M}_N &\equiv \mathbf{Z}_N^T\mathbf{M}_h\mathbf{Z}_N = ((\zeta^j,\zeta^i)_{L^2(\Omega)})_{i,j=1}^N.
\end{aligned}$$

The algebraic form of the Galerkin-ROM problem is as follows: Given $\boldsymbol{\mu} \in \mathcal{P}$, find $\mathbf{u}_N(t;\mu) \in \mathbb{R}^N$, $t \in \mathcal{I}$, such that

$$\mathbf{M}_N \frac{d\mathbf{u}_N(t;\boldsymbol{\mu})}{dt} + \mathbf{r}_N(\mathbf{u}_N(t;\boldsymbol{\mu});\boldsymbol{\mu}) = 0 \quad \text{in } \mathbb{R}^N, \tag{6.15}$$

and $\mathbf{u}_N(t=0;\boldsymbol{\mu}) = \mathbf{u}_N^0(\boldsymbol{\mu})$, where $\mathbf{u}_N^0(\boldsymbol{\mu}) \in \mathbb{R}^N$ is the generalized coordinate for $\Pi_N u^0(\boldsymbol{\mu})$. We then evaluate the output $s_N(t;\boldsymbol{\mu}) = \mathbf{q}_N(\mathbf{u}_N(t;\boldsymbol{\mu});\boldsymbol{\mu})$. The operator form (6.14) and the algebraic form (6.15) are equivalent in the sense that $u_N(t;\boldsymbol{\mu}) = \sum_{j=1}^N \mathbf{u}_N^j(t;\boldsymbol{\mu})\zeta^j$.

Most aerodynamics shape optimization and flight-parameter sweep scenarios consider steady-state solutions. The steady-state problem seeks $\mathbf{u}_N(\boldsymbol{\mu}) \in V_N$ such that

$$\mathbf{r}_N(\mathbf{u}_N(\boldsymbol{\mu});\boldsymbol{\mu}) = 0 \quad \text{in } \mathbb{R}^N, \tag{6.16}$$

and then evaluates $s_N(\boldsymbol{\mu}) \equiv \mathbf{q}_N(\mathbf{u}_N(\boldsymbol{\mu});\boldsymbol{\mu})$.

We make a few observations. First, the reduced-order Galerkin problem (6.14) (or (6.15)) is in semi-discrete form; as described for FOMs in Section 6.2.3, we apply a suitable time marching scheme to obtain a full-discrete form of the Galerkin-ROM problem. Second, the steady-state problem (6.16) is solved using a pseudo-time continuation method as discussed for FOMs in Section 6.2.3, and hence the unsteady equations are relevant also for steady-state problems. Third, although the approximation space V_N is of dimension N, the computation of the reduced residual $\mathbf{r}_N(\mathbf{w}_N;\boldsymbol{\mu}) = \mathbf{Z}_N^T\mathbf{r}_h(\mathbf{Z}_N\mathbf{w}_N;\boldsymbol{\mu})$ requires $\mathcal{O}(N_h) \gg \mathcal{O}(N)$ operations, because the FOM residual $\mathbf{r}_h(\mathbf{Z}_N\mathbf{w}_N;\boldsymbol{\mu}) \in \mathbb{R}^{N_h}$ must be projected onto the reduced basis $\mathbf{Z}_N \in \mathbb{R}^{N_h\times N}$. Hyperreduction, which enables $\mathcal{O}(N)$ evaluation of the residual, is discussed in Section 6.4.2.

6.4.1.2 Minimum-residual method

We now discuss an alternative projection method: the minimum-residual method. As the name suggests, we choose the element of V_N that minimizes the (dual) norm of the residual as our ROM solution. For steady problems, the minimum-residual problem is as follows: Given $\boldsymbol{\mu} \in \mathcal{P}$, find $u_N(\boldsymbol{\mu}) \in V_N$ such that

$$u_N(\boldsymbol{\mu}) = \arg\inf_{w_N\in V_N}\|r_h(u_N(\boldsymbol{\mu});\boldsymbol{\mu})\|_{V_h'} \equiv \arg\inf_{w_N\in V_N}\sup_{v_h\in V_h}\frac{\langle r_h(u_N(\boldsymbol{\mu});\boldsymbol{\mu}),v_h\rangle}{\|v_h\|_{V_h}}. \tag{6.17}$$

An algebraic form of the problem is as follows: Given $\boldsymbol{\mu} \in \mathcal{P}$, find $\mathbf{u}_N(\boldsymbol{\mu}) \in \mathbb{R}^N$ such that

$$\begin{aligned} \mathbf{u}_N(\mu) &= \arg\inf_{\mathbf{w}_N \in \mathbb{R}^N} \|\mathbf{r}_h(\mathbf{Z}_N \mathbf{w}_N; \boldsymbol{\mu})\|^2_{\mathbf{W}_h} \\ &= \arg\inf_{\mathbf{w}_N \in \mathbb{R}^N} \mathbf{r}_h(\mathbf{Z}_N \mathbf{w}_N; \boldsymbol{\mu})^T \mathbf{W}_h \mathbf{r}_h(\mathbf{Z}_N \mathbf{w}_N; \boldsymbol{\mu}), \end{aligned} \tag{6.18}$$

where $\mathbf{W}_h \in \mathbb{R}^{N_h \times N_h}$ is the inner product matrix; the choice $\mathbf{W}_h = \mathbf{V}_h^{-1}$ for $\mathbf{V}_{h,ij} = (\varphi^j, \varphi^i)_{V_h}$, $i, j = 1, \ldots, N_h$, results in (6.18) to be equivalent to (6.17).

The minimum-residual formulation can also be extended to unsteady problems as follows: Given $\boldsymbol{\mu} \in \mathcal{P}$, find $\mathbf{u}_N(t; \boldsymbol{\mu}) \in \mathbb{R}^N$, $t \in I$, such that

$$\mathbf{u}_N^k(\mu) = \arg\inf_{\mathbf{w}_N \in \mathbb{R}^N} \|\mathbf{r}_{h,\Delta t}(\mathbf{Z}_N \mathbf{w}_N; \{\mathbf{Z}_N \mathbf{u}_N^l(\boldsymbol{\mu})\}_{l=1}^{k-1}; \boldsymbol{\mu})\|_{\mathbf{W}_h}, \quad k = 1, \ldots, K.$$

The formulation minimizes the residual associated with each time step.

We make a few observations. First, the minimum-residual method can be cast as a Petrov–Galerkin method [50]; as a result, the method is also referred to as a least-squares Petrov–Galerkin method [21]. Second, the minimum-residual method is a very common approach for model reduction of steady nonlinear aerodynamics problems and has been used in works including [43, 44, 45, 69, 80, 79]. Third, similarly to the Galerkin method, the evaluation of the FOM residual in (6.18) requires $\mathcal{O}(N^h) \gg \mathcal{O}(N)$ operations. Hyperreduction, which enables $\mathcal{O}(N)$ evaluation of the residual, is discussed in Section 6.4.2.

6.4.1.3 Other approaches: interpolation- and L^1-based ROMs

While the Galerkin and minimum-residual methods are most commonly used methods for model reduction of nonlinear aerodynamics problems, some works have used interpolation-based ROMs, which deduce the reduced basis coefficients $\mathbf{u}_N \in \mathbb{R}^N$ through interpolation. In the context of aerodynamics, the approach has been applied to flight-parameter sweep scenarios: Bui-Thanh et al. [16] deduce the reduced basis coefficients using cubic splines for two-dimensional Euler flow over an airfoil; Franz et al. [30] deduce the reduced basis coefficients using a manifold learning technique for three-dimensional Euler flow over a wing.

We can also consider minimization of different norms of the residual to deduce $\mathbf{u}_N \in \mathbb{R}^N$. Of particular interest is the L^1-norm, which is a more natural norm for hyperbolic equations. Based on this observation, Abgrall and Crisovan [1] propose an ROM which identifies the solution through L^1-minimization and apply it to parameterized transonic Euler flow over an airfoil.

6.4.2 Hyperreduction

As discussed in Section 6.4.1, seeking the solution in a reduced space $V_N \subset V_h$ is insufficient to achieve $\mathcal{O}(N^\bullet)$ online cost for nonlinear problems. We need a means to approximate the projection of the FOM residual $\mathbf{r}_h(\mathbf{w}_N;\boldsymbol{\mu}) \in \mathbb{R}^{N_h}$ onto the reduced basis $\mathbf{Z}_N \in \mathbb{R}^{N_h \times N}$ in $\mathcal{O}(N)$ operations for the Galerkin method, and there is an analogous requirement for the minimum-residual method. This is the goal of hyperreduction, a term coined by Ryckelynck [58]. We here present hyperreduction approaches that have been used for aerodynamics problems; we refer to Chapter 5 of Volume 2 for a more general coverage. We follow the convention used in much of the hyperreduction literature and present formulations in algebraic form.

6.4.2.1 Minimum-residual collocation methods

We first consider arguably the simplest hyperreduction method: the minimum-residual method with a collocation-based approximation of the residual norm. To begin, we assume that the FOM residual can be decomposed into elemental contributions; the assumption holds for finite volume and finite element methods – the two most commonly used discretizations in aerodynamics – as the FOM residual is assembled element by element. We express this elemental decomposition of the residual as

$$\mathbf{r}_h(\mathbf{w}_h;\boldsymbol{\mu}) = \sum_{\kappa=1}^{N_e} \mathbf{r}_{h,\kappa}(\mathbf{w}_h;\boldsymbol{\mu}) \quad \text{in } \mathbb{R}^{N_h},$$

where $N_e \equiv |\mathcal{T}_h|$ is the number of elements and $\mathbf{r}_{h,\kappa} : \mathbb{R}^{N_h} \times \mathcal{P} \to \mathbb{R}^{N_h}$ is the FOM residual operator for the κ-th element. Note that $\mathbf{r}_{h,\kappa}(\mathbf{w}_h;\boldsymbol{\mu}) \in \mathbb{R}^{N_h}$ is mostly sparse, because a given element contributes to a small number of residual degrees of freedom.

We now proceed with hyperreduction. We first choose a small subset of $\tilde{N}_e$ sample elements $\tilde{\mathcal{T}}_h \subset \mathcal{T}_h$ so that $N \le \tilde{N}_e \ll N_e$; we denote the associated sample element indices by $\tilde{T}$. (Quantities associated with hyperreduction bear $\tilde{\cdot}$ throughout this section.) We then consider the following hyperreduced approximation of the minimum-residual problem (6.18): Given $\boldsymbol{\mu} \in \mathcal{P}$, find $\tilde{\mathbf{u}}_N(\boldsymbol{\mu}) \in \mathbb{R}^N$ such that

$$\tilde{\mathbf{u}}_N(\boldsymbol{\mu}) = \underset{\mathbf{w}_N \in \mathbb{R}^N}{\arg\min} \left\| \sum_{\kappa \in \tilde{\mathcal{T}}_h} \mathbf{r}_{h,\kappa}(\mathbf{w}_h;\boldsymbol{\mu}) \right\|_2 . \tag{6.19}$$

We observe that if $\tilde{N}_e = \mathcal{O}(N) \ll N_h$, then we can solve this hyperreduced minimum-residual problem in $\mathcal{O}(N^\bullet)$ operations.

We can also describe the hyperreduction procedure algebraically. To this end, we first identify the set of $\tilde{N}_{\tilde{\mathcal{I}}}$ residual sample indices $\tilde{\mathcal{I}} \equiv \{\tilde{i}_1, \dots, \tilde{i}_{\tilde{N}_{\tilde{\mathcal{I}}}}\}$ associated with the sample elements $\tilde{\mathcal{T}}_h$. For finite volume methods, $\tilde{N}_{\tilde{\mathcal{I}}} = N_c \tilde{N}_e$ as the number of residual degrees of freedom associated with each element is equal to the number of components N_c in the PDE. We then introduce the associated sample matrix $\mathbf{P} = (e^{i_1}, \ldots, e^{i_{\tilde{N}_{\tilde{\mathcal{I}}}}}) \in \mathbb{R}^{N_h \times \tilde{N}_{\tilde{\mathcal{I}}}}$ whose j-th column is the canonical unit vector $e^{i_j} \in \mathbb{R}^{N_h}$. The minimum-residual collocation problem (6.19) is equivalent to

$$\tilde{\mathbf{u}}_N(\boldsymbol{\mu}) = \arg\min_{\mathbf{w}_N \in \mathbb{R}^N} \|\mathbf{P}^T \mathbf{r}_h(\mathbf{w}_h; \boldsymbol{\mu})\|_2.$$

Here, to achieve hyperreduction, we evaluate the operator $(\mathbf{P}^T \mathbf{r}_h) : \mathbb{R}^{N_h} \times \mathcal{P} \to \mathbb{R}^{\tilde{N}_{\tilde{\mathcal{I}}}}$ by first checking which indices are requested by $\mathbf{P}$ and then computing the residual for only those indices.

The key to a successful hyperreduction by the minimum-residual collocation formulation lies in the selection of the sample elements $\tilde{\mathcal{T}}_h$, which is performed in the offline stage. We here review a few approaches that have been applied to aerodynamics problems.

Physics-informed selection. To our knowledge, LeGresley and Alonso [44] were the first to consider hyperreduction for aerodynamics problems. In the work, hyperreduction is achieved by including only 20 %–30 % of the elements near the airfoil in $\tilde{\mathcal{T}}_h$. This strategy was specialized for aerodynamic shape optimization, in which most of the solution variations are in the vicinity of the airfoil. Vendl et al. [66] also consider a physics-informed hyperreduction in the context of flight-parameter sweep; however, as the parameter affects the boundary conditions, they also included elements on the far-field boundary in $\tilde{\mathcal{T}}_h$.

Gappy POD on the state snapshots. To devise a more systematic approach to identify sample elements, Washabaugh et al. [69, 70] invoke gappy POD [29] on the solution snapshots $\mathbf{S} \equiv (\mathbf{u}_h(\boldsymbol{\mu}^1), \ldots, \mathbf{u}_h(\boldsymbol{\mu}^{N_s})) \in \mathbb{R}^{N_h \times N_s}$ and set the sample indices $\tilde{\mathcal{I}}$ for the minimum-residual collocation method equal to the gappy POD sample indices. Specifically, the method successively processes sets of snapshots $\mathbf{S} \in \mathbb{R}^{N_h \times N_s}$ in smaller batches $\mathbf{S}_k = (\mathbf{u}_h(\boldsymbol{\mu}^1), \ldots, \mathbf{u}_h(\boldsymbol{\mu}^k)) \in \mathbb{R}^{N_h \times k}$, $k = 1, \ldots, N_s$; assuming the sample indices $\tilde{\mathcal{I}}$ have been constructed for $\tilde{\mathbf{S}}_{k-1}$, the sample indices are updated for the batch $\tilde{\mathbf{S}}_k$ as follows:

1. Compute the gappy POD reconstruction of the snapshots:
   $\tilde{\mathbf{S}}_k = \mathbf{Z}_N(\mathbf{P}^T \mathbf{Z}_N)^{\dagger} \mathbf{P}^T \mathbf{S} \in \mathbb{R}^{N_h \times N_s}$. Here, $(\cdot)^{\dagger}$ denotes the pseudo-inverse.
2. Set $i^\star = \arg\max_{i \in [1,N_h]} \max_{j \in [1,N_s]} |\mathbf{S}_k - \tilde{\mathbf{S}}_k|_{ij}$.
3. Add the sample index: $\tilde{\mathcal{I}} = \tilde{\mathcal{I}} \cup i^\star$; update the sample matrix $\mathbf{P}$ accordingly.

This approach assumes that the sample indices with which the state can be approximated work well also for the residual; this assumption allows the method to work with the state and not the residual, which significantly reduces the offline cost relative to Gauss–Newton approximate tensor (GNAT) and empirical quadrature procedure (EQP) methods discussed in Sections 6.4.2.2 and 6.4.2.3, respectively. The method has been applied to full aircraft configuration under shape deformations [69] as discussed further in Section 6.4.5.

6.4.2.2 Gauss–Newton approximate tensor method

The GNAT method [21, 22] approximates the minimum-residual problem (6.18) for $\mathbf{W}_h = \mathbf{I}$, $\mathbf{u}_N(\boldsymbol{\mu}) = \arg\inf_{\mathbf{w}_N\in\mathbb{R}^N} \|\mathbf{r}_h(\mathbf{Z}_N\mathbf{w}_N;\boldsymbol{\mu})\|_2$, using a gappy POD approximation [29] of the residual and Jacobian and then solves the problem using the Gauss–Newton method. (Although the original work [21, 22] considers unsteady problems, for notational simplicity we here consider a steady problem.) The solution of (6.18) by the Gauss–Newton method requires successive solution of the linear least-squares problem: Find the update $\delta\mathbf{w}_N \in \mathbb{R}^N$ such that

$$\delta\mathbf{w}_N = \arg\min_{\mathbf{v}_n\in\mathbb{R}^N}\big\|\mathbf{J}_h(\mathbf{Z}_N\mathbf{v}_N)\mathbf{Z}_N\mathbf{v}_N + \mathbf{r}_h(\mathbf{Z}_N\mathbf{w}_N)\big\|_2. \tag{6.20}$$

The solution is then updated according to $\mathbf{w}_N \leftarrow \mathbf{w}_N + \alpha\delta\mathbf{w}_N$, where the step length $\alpha \in (0,1]$ is deduced by line search. The cost to solve this least-squares problem is $\mathcal{O}(N_h)$ as it requires the FOM residual and Jacobian.

To approximately solve (6.20) in $\mathcal{O}(N)$ operations, the GNAT method prepares three ingredients for a gappy POD approximation of the residual $\mathbf{r}_h : \mathbb{R}^{N_h}\times\mathcal{P} \to \mathbb{R}^{N_h}$: (i) a reduced basis for the residual $\mathbf{Z}^r \in \mathbb{R}^{N_h\times N_r}$, (ii) a set of sample indices $\mathcal{I} = \{i_1,\dots,i_{\tilde{N}_{\tilde{\mathcal{I}}}}\}$ for $\tilde{N}_{\tilde{\mathcal{I}}} \geq N_r$, and (iii) the associated sample matrix $\mathbf{P} = (e^{i_1},\dots,e^{i_{\tilde{N}_{\tilde{\mathcal{I}}}}}) \in \mathbb{R}^{N_h\times\tilde{N}_{\tilde{\mathcal{I}}}}$ whose j-column is the canonical unit vector $e^{i_j} \in \mathbb{R}^{N_h}$. The residual is then approximated by regression: $\tilde{\mathbf{r}}_h(\mathbf{Z}_N\mathbf{w}_N) = \arg\min_{\mathbf{v}\in\mathbf{v}_r} \|\mathbf{P}^T(\mathbf{r}_h(\mathbf{Z}_N\mathbf{w}_N) - \mathbf{Z}^r\mathbf{v})\|_2$. The Jacobian is similarly approximated using a reduced basis for the Jacobian $\mathbf{Z}^J \in \mathbb{R}^{N_h\times N_J}$ and the same sample matrix $\mathbf{P}^T$ by regression: $\tilde{\mathbf{J}}_h(\mathbf{Z}_N\mathbf{w}_N)\mathbf{Z}_{N,j} = \arg\min_{\mathbf{v}\in\mathbf{v}_r} \|\mathbf{P}^T(\mathbf{J}_h(\mathbf{Z}_N\mathbf{w}_N)\mathbf{Z}_{N,j} - \mathbf{Z}^J\mathbf{v})\|_2$, $j = 1,\dots,N$. The GNAT method solves this gappy POD-approximated minimum-residual problem using a gappy POD-approximated Gauss–Newton method.

Carlberg et al. [21, 22] introduce four variants of the GNAT method, named procedure 0–3. We here consider only procedure 1, which has been shown to exhibit good accuracy and robustness for unsteady aerodynamics problems. We outline the offline and online stages of the GNAT method.

Offline stage. In the offline stage, we construct all ingredients of GNAT: $\mathbf{Z}_N \in \mathbb{R}^{N_h\times N_r}$, $\mathbf{Z}^r = \mathbf{Z}^J \in \mathbb{R}^{N_h\times N_r}$, and $\mathbf{P} \in \mathbb{R}^{N_h\times\tilde{N}_{\tilde{\mathcal{I}}}}$.

1. Choose a snapshot parameter set $\Xi_\mathrm{t} = \{\boldsymbol{\mu}^i\}_{i=1}^{N_\mathrm{t}} \subset \mathcal{P}$.
2. Solve the FOM (6.4) for each $\boldsymbol{\mu} \in \Xi_\mathrm{t}$ to obtain $\{\mathbf{u}_h(\boldsymbol{\mu})\}_{\boldsymbol{\mu}\in\Xi_\mathrm{t}}$. Apply POD to the snapshots to obtain a state reduced basis $\mathbf{Z}_N \in \mathbb{R}^{N_h\times N}$.
3. Solve the nonhyperreduced ROM (6.17) for each $\boldsymbol{\mu} \in \Xi_\mathrm{t}$. Collect residual snapshots $\{\mathbf{r}_h(\mathbf{Z}_N\mathbf{u}_N(\boldsymbol{\mu});\boldsymbol{\mu})\}_{\boldsymbol{\mu}\in\Xi_s}$. Apply POD to the set to obtain a residual reduced basis $\mathbf{Z}^r \in \mathbb{R}^{N_h\times N_r}$ for $N_r \geq N$. Set $\mathbf{Z}^J = \mathbf{Z}^r$.
4. Apply the gappy POD procedure described in Section 6.4.2.1 (for the state snapshots) to the residual snapshots to determine the sample index $\tilde{\mathcal{I}}$ with $\tilde{N}_{\tilde{\mathcal{I}}} \geq N_r$ and the associated sample matrix $\mathbf{P} \in \mathbb{R}^{N_h\times\tilde{N}_{\tilde{\mathcal{I}}}}$.
5. Precompute $\mathbf{A} \equiv (\mathbf{P}^T\mathbf{Z}^J)^\dagger \in \mathbb{R}^{N_J\times N_{\tilde{\mathcal{I}}}}$ and $\mathbf{B} \equiv (\mathbf{Z}^J)^T\mathbf{Z}^r(\mathbf{P}^T\mathbf{Z}^r)^\dagger \in \mathbb{R}^{N_J\times N_{\tilde{\mathcal{I}}}}$.

Online stage. In the online stage, given $\boldsymbol{\mu} \in \mathcal{P}$, we seek $\mathbf{u}_N(\boldsymbol{\mu})$ such that

$$\mathbf{u}_N(\boldsymbol{\mu}) = \arg\min_{\mathbf{w}_N \in \mathbb{R}^N} \|\mathbf{Z}^r(\mathbf{P}^T\mathbf{Z}^r)^\dagger \mathbf{P}^T \mathbf{r}_h(\mathbf{Z}_N\mathbf{w}_N; \boldsymbol{\mu})\|_2.$$

This problem is solved using the Gauss–Newton method as follows:
1. Form $\mathbf{C}(\mathbf{w}_N) = \mathbf{P}^T\mathbf{J}_h(\mathbf{Z}_N\mathbf{w}_N)\mathbf{Z}_N$ and $\mathbf{D}(\mathbf{w}_N) = \mathbf{P}^T\mathbf{r}_h(\mathbf{Z}_N\mathbf{w})$.
2. Solve the linear least-squares problem: Find $\delta\mathbf{w}_N \in \mathbb{R}^N$ such that

$$\delta\mathbf{w}_N = \arg\min_{\mathbf{v}\in\mathbb{R}^N} \|\mathbf{A}\mathbf{C}(\mathbf{w}_N)\mathbf{v} + \mathbf{B}\mathbf{D}(\mathbf{w}_N)\|_2.$$

3. Update $\mathbf{w}_N \leftarrow \mathbf{w}_N + \alpha\delta\mathbf{w}_N$, where α is determined from line search.
4. If converged, terminate; otherwise return to 1.

The online computational cost is $\mathcal{O}(N^\bullet)$ and is independent of the FOM. To evaluate the output, the GNAT method does not explicitly hyperreduce the output functional $\mathbf{q}_h : \mathbb{R}^{N_h} \times \mathcal{P} \to \mathbb{R}$, but simply leverages the fact that output functionals for most aerodynamics problems require evaluation on a small subset of elements, e. g., elements on aerodynamics surfaces. Hence, output evaluation constitutes a small fraction of the overall cost.

The GNAT method has been applied to large-scale simulation of (nonparameterized) unsteady turbulent flow over the Amhed body [21, 22] as discussed further in Section 6.4.5. We also refer to [20] for a detailed analysis of the method.

6.4.2.3 Galerkin method with empirical quadrature procedure

One of the limitations of the hyperreduction methods discussed in the previous two sections is that they do not provide a quantitative control of the solution and/or output error due to hyperreduction. One approach which provides such quantitative error control is the EQP [77, 75, 76]. To describe the method, we first introduce the hyperreduced residual $\tilde{\mathbf{r}}_N : \mathbb{R}^N \times \mathcal{P} \to \mathbb{R}^N$ and output functional $\tilde{\mathbf{q}}_N : \mathbb{R}^N \times \mathcal{P} \to \mathbb{R}$ of the form

$$\tilde{\mathbf{r}}_N(\mathbf{w}_N; \boldsymbol{\mu}) \equiv \sum_{\kappa=1}^{N_e} \rho_\kappa^r \mathbf{r}_{N,\kappa}(\mathbf{w}_N; \boldsymbol{\mu}) \equiv \sum_{\kappa=1}^{N_e} \rho_\kappa^r \mathbf{Z}_N^T \mathbf{r}_{h,\kappa}(\mathbf{Z}_N\mathbf{w}_N; \boldsymbol{\mu}), \tag{6.21}$$

$$\tilde{\mathbf{q}}_N(\mathbf{w}_N; \boldsymbol{\mu}) \equiv \sum_{\kappa=1}^{N_e} \rho_\kappa^q \mathbf{q}_{N,\kappa}(\mathbf{w}_N; \boldsymbol{\mu}) \equiv \sum_{\kappa=1}^{N_e} \rho_\kappa^q \mathbf{q}_{h,\kappa}(\mathbf{Z}_N\mathbf{w}_N; \boldsymbol{\mu}); \tag{6.22}$$

here $\rho^r \in \mathbb{R}^{N_e}$ and $\rho^q \in \mathbb{R}^{N_e}$ are the EQP weights that are sparse (i. e., most entries are zero) so that the summands need to be evaluated for a small subset of elements. The associated hyperreduced problem is as follows: Given $\boldsymbol{\mu} \in \mathcal{P}$, find $\tilde{\mathbf{u}}_N(t;\boldsymbol{\mu}) \in \mathbb{R}^N$, $t \in \mathcal{I}$, such that

$$\mathbf{M}_N \frac{d\tilde{\mathbf{u}}_N(t;\boldsymbol{\mu})}{dt} + \tilde{\mathbf{r}}_N(\tilde{\mathbf{u}}_{N,M}(t;\boldsymbol{\mu});\boldsymbol{\mu}) = 0 \quad \text{in } \mathbb{R}^N,$$

for $\tilde{\mathbf{u}}_N(t=0;\boldsymbol{\mu}) = \mathbf{u}_N^0(\boldsymbol{\mu})$, and evaluate the output $\tilde{s}_N(t;\boldsymbol{\mu}) = \tilde{\mathbf{q}}_N(\tilde{\mathbf{u}}_N(t;\boldsymbol{\mu});\boldsymbol{\mu})$. We wish to find EQP weights $\rho^r \in \mathbb{R}^{N_e}$ and $\rho^q \in \mathbb{R}^{N_e}$ so that (i) $|s_N(t;\boldsymbol{\mu}) - \tilde{s}_N(t;\boldsymbol{\mu})| \le \delta$ for a user-prescribed tolerance $\delta \in \mathbb{R}_{>0}$ and (ii) $\mathrm{nnz}(\rho^r) = \mathcal{O}(N)$ and $\mathrm{nnz}(\rho^q) = \mathcal{O}(N)$. The two conditions ensure the accuracy and online efficiency, respectively, of the hyperreduced ROM.

The EQP weights are computed in the offline stage by solving linear programs (LPs). We first introduce a parameter training set $\Xi_\mathrm{t} \equiv \{\hat{\boldsymbol{\mu}}^j\}_{j=1}^{N_t}$ and the associated training states $U_\mathrm{t} = \{\hat{\mathbf{u}}_N(\boldsymbol{\mu})\}_{\boldsymbol{\mu}\in\Xi_\mathrm{t}}$. The training states can be the nonhyperreduced ROM solution as it is done for GNAT; however, when used in conjunction with the greedy algorithm, U_t can be the hyperreduced ROM solution in a given iteration [75]. The general form of the linear program, denoted $\mathsf{LP}^\bullet$, where $^\bullet$ is the placeholder for the residual "r" or output function "q," is as follows: Find the basic feasible solution $\rho^{\bullet,\star} \in \mathbb{R}^{N_e}$ such that

$$\rho^{\bullet,\star} = \arg\min_{\rho^\bullet \in \mathbb{R}^{N_e}} \sum_{\kappa=1}^{N_e} \rho_\kappa^\bullet,$$

subject to nonnegativity constraints

$$\rho_\kappa^\bullet \ge 0, \quad \kappa = 1,\dots,N_e,$$

and manifold-accuracy and constant-integration constraints

$$\begin{pmatrix} \mathbf{a}_1^\bullet(\boldsymbol{\mu}^1) & \cdots & \mathbf{a}_{N_e}^\bullet(\boldsymbol{\mu}^1) \\ \vdots & \ddots & \vdots \\ \mathbf{a}_1^\bullet(\boldsymbol{\mu}^{N_\mathrm{t}}) & \cdots & \mathbf{a}_{N_e}^\bullet(\boldsymbol{\mu}^{N_\mathrm{t}}) \\ \hline |\kappa_1| & \cdots & |\kappa_{N_e}| \end{pmatrix} \begin{pmatrix} \rho_1^\bullet \\ \vdots \\ \rho_{N_e}^\bullet \end{pmatrix} \lessgtr \begin{pmatrix} \mathbf{b}^\bullet(\boldsymbol{\mu}^1) \\ \vdots \\ \mathbf{b}^\bullet(\boldsymbol{\mu}^{N_\mathrm{t}}) \\ \hline |\Omega| \end{pmatrix} \pm \begin{pmatrix} \boldsymbol{\delta}^\bullet \\ \vdots \\ \boldsymbol{\delta}^\bullet \\ \hline \delta_\Omega \end{pmatrix}, \tag{6.23}$$

where $\mathbf{a}_\kappa(\boldsymbol{\mu}) \in \mathbb{R}^{N_c^\bullet}$, $\kappa = 1,\dots,N_e$, is a set of vectors that depends on the specific manifold accuracy constraint to be described shortly, $N_c^\bullet$ is the number of constraints per training parameter, $b^\bullet(\boldsymbol{\mu}) \equiv \sum_{\kappa=1}^{N_e} \mathbf{a}_\kappa^\bullet(\boldsymbol{\mu}) \in \mathbb{R}^{N_c^\bullet}$, $\boldsymbol{\delta}^\bullet \in \mathbb{R}^{N_c^\bullet}$ is the manifold-accuracy tolerance, $|\kappa| \equiv \int_\kappa dx$, and $|\Omega| \equiv \int_\Omega dx$. The LP can be solved using a simplex method. We now introduce specific manifold accuracy constraints for the residual (6.21) and output functional (6.22).

Residual EQP. The residual EQP weights $\rho^r \in \mathbb{R}^{N_e}$ are found by solving $\mathsf{LP}^r(\Xi_t, U_t, \delta^r)$. As our goal is to control the output error, we introduce a reduced basis approximation of the dual problem: Given $\boldsymbol{\mu} \in \mathcal{P}$ and the linearization state $\mathbf{u}_N(\boldsymbol{\mu}) \in \mathbb{R}^N$, find the dual solution $\mathbf{z}_N(\boldsymbol{\mu}) \in \mathbb{R}^N$ such that

$$\mathbf{J}_N(\hat{\mathbf{u}}_N(\boldsymbol{\mu});\boldsymbol{\mu})^T \mathbf{z}_N(\boldsymbol{\mu}) = \mathbf{g}_N(\hat{\mathbf{u}}_N(\boldsymbol{\mu});\boldsymbol{\mu}) \quad \text{in } \mathbb{R}^N.$$

As discussed in the context of balanced POD in Section 6.3.2.4, the dual solution relates the residual to the output error. The manifold-accuracy constraint (6.23) for the residual imposes $N_c^r = N$ constraints per training parameter given by

$$\mathbf{a}_\kappa^r(\boldsymbol{\mu}) \equiv |\mathbf{z}_N(\boldsymbol{\mu})| \circ |\mathbf{r}_{N,\kappa}(\hat{\mathbf{u}}(\boldsymbol{\mu});\boldsymbol{\mu})| \quad \text{in } \mathbb{R}^N,$$

and $\boldsymbol{\delta}^r = \frac{\delta^r}{2}\mathbf{1}_N$, where $\mathbf{1}_N \in \mathbb{R}^N$ is the vector of all ones and $\circ$ is the Hadamard (i. e., entrywise) product. Overall, LP^r has N_e unknowns, N_e nonnegativity constraints, and $2(N_t N + 1)$ inequality constraints (where the leading factor of two accounts for the upper and lower bounds in (6.23)).

Output functional EQP. The output EQP weights $\rho^q \in \mathbb{R}^{N_e}$ are similarly found by solving $\mathsf{LP}^q(\Xi_t, U_t, \delta^q)$. The manifold-accuracy constraint (6.23) for the output functional imposes $N_c^q = 1$ constraint per training parameter given by

$$\mathbf{a}_\kappa^q(\boldsymbol{\mu}) \equiv \mathbf{q}_\kappa(\tilde{\mathbf{u}}_N(\boldsymbol{\mu});\boldsymbol{\mu}) \quad \text{in } \mathbb{R}.$$

Overall, LP^q has N_e unknowns, N_e nonnegativity constraints, and $2(N_t + 1)$ inequality constraints; the LP for the output functional is much smaller than that for the residual.

Output a posteriori error estimate. The EQP method also provides an a posteriori error estimate for the output error. The error estimate is based on the dual-weighted residual method [14]. To this end, we first introduce a separate reduced basis for the dual problem $\mathbf{Z}_N^{\mathrm{du}} \in \mathbb{R}^{N_h \times N}$, which is different from the primal reduced basis $\mathbf{Z}_N \in \mathbb{R}^{N_h \times N}$. We then introduce an EQP approximation of the residual, Jacobian, and output gradient evaluated with respect to the dual reduced basis $\mathbf{Z}_N^{\mathrm{du}}$: $\tilde{\mathbf{r}}_N^{\mathrm{du}} : \mathbb{R}^N \times \mathcal{P} \to \mathbb{R}^N$, $\tilde{\mathbf{J}}_N^{\mathrm{du}} : \mathbb{R}^N \times \mathcal{D} \to \mathbb{R}^{N \times N}$ and $\tilde{\mathbf{g}}_N^{\mathrm{du}} : \mathbb{R}^N \times \mathcal{D} \to \mathbb{R}^N$ such that

$$\begin{aligned}
\tilde{\mathbf{r}}_N^{\mathrm{du}}(\mathbf{w};\boldsymbol{\mu}) &\equiv \sum_{\kappa=1}^{N_e} \rho_\kappa^\eta \mathbf{r}_{N,\kappa}^{\mathrm{du}}(\mathbf{Z}_N \mathbf{w}_N;\boldsymbol{\mu}) \equiv \sum_{\kappa=1}^{N_e} \rho_\kappa^\eta \mathbf{Z}_N^{\mathrm{du}\,T} \mathbf{r}_{h,\kappa}(\mathbf{Z}_N \mathbf{w}_N;\boldsymbol{\mu}), \\
\tilde{\mathbf{J}}_N^{\mathrm{du}}(\mathbf{w};\boldsymbol{\mu}) &\equiv \sum_{\kappa=1}^{N_e} \rho_\kappa^\eta \mathbf{J}_{N,\kappa}^{\mathrm{du}}(\mathbf{Z}_N \mathbf{w}_N;\boldsymbol{\mu}) \equiv \sum_{\kappa=1}^{N_e} \rho_\kappa^\eta \mathbf{Z}_N^{\mathrm{du}\,T} \mathbf{J}_{h,\kappa}(\mathbf{Z}_N \mathbf{w}_N;\boldsymbol{\mu}) \mathbf{Z}_N^{\mathrm{du}}, \\
\tilde{\mathbf{g}}_N^{\mathrm{du}}(\mathbf{w};\boldsymbol{\mu}) &\equiv \sum_{\kappa=1}^{N_e} \rho_\kappa^\eta \mathbf{g}_{N,\kappa}(\mathbf{Z}_N \mathbf{w}_N;\boldsymbol{\mu}) \equiv \sum_{\kappa=1}^{N_e} \rho_\kappa^\eta \mathbf{Z}_N^{\mathrm{du}\,T} \mathbf{g}_{h,\kappa}(\mathbf{Z}_N \mathbf{w}_N;\boldsymbol{\mu}),
\end{aligned}$$

for some EQP weights $\rho^\eta \in \mathbb{R}^{N_e}$ computed in the offline stage. The EQP dual problem is as follows: Given $\boldsymbol{\mu} \in \mathcal{D}$ and $\tilde{\mathbf{u}}_N(\boldsymbol{\mu}) \in \mathbb{R}^N$, find $\tilde{\mathbf{z}}_N^{\mathrm{du}}(\boldsymbol{\mu}) \in \mathbb{R}^N$ such that

$$\tilde{\mathbf{J}}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N(\boldsymbol{\mu});\boldsymbol{\mu})^T \tilde{\mathbf{z}}_N^{\mathrm{du}}(\boldsymbol{\mu}) = \tilde{\mathbf{g}}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N(\boldsymbol{\mu});\boldsymbol{\mu}) \quad \text{in } \mathbb{R}^N.$$

The output error estimate is given by

$$\tilde{\eta}_N^{\mathrm{rb}}(\boldsymbol{\mu}) \equiv \left|\tilde{\mathbf{z}}_N^{\mathrm{du}}(\boldsymbol{\mu})^T \tilde{\mathbf{r}}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N(\boldsymbol{\mu});\boldsymbol{\mu})\right|.$$

Assuming $\mathrm{nnz}(\rho^\eta) = \mathcal{O}(N)$, this error estimate is computable in $\mathcal{O}(N)$ operations.

The output error estimate EQP weights $\rho^\eta \in \mathbb{R}^{N_e}$ is given by a linear program $\mathrm{LP}^\eta(\Xi_{\mathrm{t}}, U_{\mathrm{t}}, \delta^\eta)$. The manifold-accuracy constraint (6.23) for the output error estimate imposes $N_c^\eta = 3N$ constraints per training parameter given by

$$\mathbf{a}_{N,\kappa}^\eta(\boldsymbol{\mu}) \equiv \begin{pmatrix} \max\{|\mathbf{z}_N^{\mathrm{du}}(\boldsymbol{\mu})|, \mathbf{z}_{\min}^{\mathrm{du}}\} \circ |\mathbf{r}_{N,\kappa}^{\mathrm{du}}(\boldsymbol{\mu})| \\ \max\{|\mathbf{r}_N^{\mathrm{du}}(\boldsymbol{\mu})|, \mathbf{r}_{\min}^{\mathrm{du}}\} \circ |\mathbf{J}_N^{\mathrm{du}}(\boldsymbol{\mu})^{-T} \mathbf{J}_{N,\kappa}^{\mathrm{du}}(\boldsymbol{\mu})^T \mathbf{z}_N^{\mathrm{du}}(\boldsymbol{\mu})| \\ \max\{|\mathbf{r}_N^{\mathrm{du}}(\boldsymbol{\mu})|, \mathbf{r}_{\min}^{\mathrm{du}}\} \circ |\mathbf{J}_N^{\mathrm{du}}(\boldsymbol{\mu})^{-T} \mathbf{g}_{N,\kappa}^{\mathrm{du}}(\boldsymbol{\mu})| \end{pmatrix} \quad \text{in } \mathbb{R}^{3N};$$

here $\mathbf{z}_{\min}^{\mathrm{du}} \equiv (\upsilon\boldsymbol{\delta}^\eta/N)^{1/2}/2$ and $\mathbf{r}_{\min} \equiv (\boldsymbol{\delta}^\eta/(\upsilon N))^{1/2}/4$ for $\upsilon \equiv \|\mathbf{z}_N^{\mathrm{du}}(\boldsymbol{\mu})\|_2/\|\mathbf{r}_N^{\mathrm{du}}(\boldsymbol{\mu})\|_2$, the maximum operator is taken entrywise, and all entities with the argument $\boldsymbol{\mu}$ are evaluated about the state $\tilde{\mathbf{u}}(\tilde{\boldsymbol{\mu}})$ and the parameter $\boldsymbol{\mu}$; e. g., $\mathbf{r}_{N,\kappa}^{\mathrm{du}}(\boldsymbol{\mu}) \equiv \mathbf{r}_{N,\kappa}^{\mathrm{du}}(\tilde{\mathbf{u}}_N(\boldsymbol{\mu});\boldsymbol{\mu})$. Overall, LP^η has N_e unknowns, N_e nonnegativity constraints, and $2(3N_tN + 1)$ inequality constraints.

The EQP method has been applied to two- and three-dimensional turbulent aerodynamic flows in the context of flight-parameter sweep [75, 76]. The rapidly computable output error estimate enables the construction of a reduced model that meets the user-prescribed error tolerance in an automated manner in the offline stage and provides reliable predictions in the online stage.

6.4.2.4 Choice of a hyperreduction procedure

We make a few remarks about the choice of a hyperreduction method for aerodynamics problems. One of the challenges in hyperreduction for aerodynamics is that the FOM is typically very large, with millions of degrees of freedom, and hence the offline training cost cannot be neglected in a practical engineering setting. This is unlike some classical model reduction scenarios, where the offline cost is often neglected. The other challenge is the stability; the hyperreduced system must provide time stability for unsteady simulations to produce meaningful results and for steady simulations to find solutions using the PTC procedure. There exist many examples in the literature where a hyperreduction method that works well for other nonlinear problems has been found to be insufficient for aerodynamics problems.

For instance, the missing point estimate [8] chooses the sample indices $\tilde{\mathcal{I}}$ such that the associated sample matrix $\mathbf{P}$ minimizes the condition number of $\mathbf{Z}_N^T\mathbf{P}\mathbf{P}^T\mathbf{Z}_N$; however, the method was deemed too expensive for steady aerodynamics problems in [66]. The empirical interpolation method [10, 31] and its discrete counterpart [23], which are arguably the most common hyperreduction methods, to our knowledge have seen limited use in aerodynamics; in fact, Carlberg et al. [22, 20] report temporal instability for turbulent unsteady flows. Similarly, the GNAT method, which has been used successfully for nonparameterized unsteady problems, was deemed too expensive for parameterized steady aerodynamics problems in Washabaugh [70]; we also refer to the work for detailed discussion of the choice of a hyperreduction method.

6.4.3 Construction of reduced basis

Techniques to find an appropriate reduced basis for nonlinear aerodynamics problems are largely the same as those for linearized aerodynamics problems discussed in Sections 6.3.2 and 6.3.3. By far the most popular method to generate reduced bases for nonlinear aerodynamics problems is POD [43, 44, 45, 69, 80, 79, 66, 21, 22]. For unsteady problems, the snapshots are collected for K time steps to yield $\mathbf{S} = \{\mathbf{u}_h^k \approx \mathbf{u}_h(t^k)\}_{k=1}^K$; for parameterized problems, the snapshots are collected for N_t training parameters $\Xi_\mathrm{t} \equiv \{\boldsymbol{\mu}^i\}_{i=1}^{N_\mathrm{t}}$ to yield $\mathbf{S} = \{\mathbf{u}_h(\boldsymbol{\mu})\}_{\boldsymbol{\mu}\in\Xi_\mathrm{t}}$. Given the snapshot matrix $\mathbf{S}$, the POD procedure to identify $\mathbf{Z}_N \in \mathbb{R}^{N_h\times N}$ is described in the context of linearized problems in Section 6.3.2.2. For the EQP method which provides an online efficient a posteriori error estimate, it is also possible to identify the reduced basis using the weak greedy algorithm discussed in Section 6.3.3.2 [75, 76]. We note that while the “standard” POD readily extends to nonlinear problems, some of its variants which rely on the linearity of the PDE, such as frequency-domain POD or balanced POD, do not.

6.4.4 Treatment of moving discontinuities

One of the challenges in model reduction of transonic aerodynamics problems is the treatment of shocks. The fundamental challenge is that if $\mathbf{u}_h(t;\boldsymbol{\mu})$ contains a discontinuity whose location depends on $t \in \mathcal{I}$ or $\boldsymbol{\mu} \in \mathcal{P}$, then the Kolmogorov N-width of $\{\mathbf{u}_h(t;\boldsymbol{\mu})\}_{t\in\mathcal{I},\boldsymbol{\mu}\in\mathcal{P}}$ is large and the solution manifold is not amenable to a low-dimensional approximate of the form $u_N(\boldsymbol{\mu}) = \zeta^j \mathbf{u}_N^j(\boldsymbol{\mu})$. We provide a brief overview of methods developed to address the challenge. We restrict our coverage to methods tested for multidimensional aerodynamics problems, and refer to the references in [53] and a review paper [54] for a more general coverage.

Domain decomposition. One way to address the problem is to forgo the reduction of the state over the entire domain and to only reduce solution over a portion of the domain, as proposed for transonic Euler flows by LeGresley and Alonso [45]. Namely, we first decompose the domain into two regions: (i) region $\Omega_\mathrm{rom} \subset \Omega$ over which the solution varies smoothly and hence $\{u_h(\boldsymbol{\mu})\,|_{\Omega_\mathrm{rom}}\}_{\boldsymbol{\mu}\in\mathcal{P}}$ is amenable to model reduction, and (ii) region $\Omega_\mathrm{fom} \equiv \Omega \setminus \Omega_\mathrm{rom}$ which contains moving discontinuities and hence is not amenable to model reduction. We then approximate the solution $u_h(\boldsymbol{\mu})|_{\Omega_\mathrm{rom}}$ using a reduced basis $\{\zeta^j\,|_{\Omega_\mathrm{rom}}\}_{j=1}^N$ and $u_h(\boldsymbol{\mu})|_{\Omega_\mathrm{fom}}$ using the native basis of the FOM.

Nonlinear model reduction. Another approach to address moving discontinuities is to consider nonlinear model reduction. Here, nonlinear model reduction refers to approaches that approximate the solution in not a linear space V_N but in a nonlinear space. (Nonlinear model reduction should not be confused with linear model reduction of nonlinear PDEs, which has been considered so far in this section.) Nonlinear model reduction approaches considered by both Cagniart et al. [19] and Nair and Balajewicz [53] are based on the following observation: If the snapshots can be translated in space such that the shocks are aligned, then the snapshots can be effectively compressed using a linear model reduction technique (e. g., POD). Specifically, the approach approximates the solution $u_h(\cdot;\boldsymbol{\mu}) \in V_h$ by

$$u_N(x;\boldsymbol{\mu}) = \zeta^j(x;\boldsymbol{\mu})\mathbf{u}_N^j(\boldsymbol{\mu})$$

for some $\mathbf{u}_N(\boldsymbol{\mu}) \in \mathbb{R}^N$ and a parameter-dependent basis

$$\zeta^j(x;\boldsymbol{\mu}) = u_h(y_j(x,\boldsymbol{\mu});\boldsymbol{\mu}), \quad j = 1,\dots,N,$$

where $y_j : \Omega \times \mathcal{P} \to \mathbb{R}^d$, $j = 1,\dots,N$, are parameter-dependent translation functions. The translation functions $\{y_j\}$ are trained in the offline stage such that the shock locations for the translated basis $\zeta^j(\cdot;\boldsymbol{\mu}) = u_h(y_j(\cdot,\boldsymbol{\mu}))$ are (approximately) aligned with the shock in $u_h(\cdot;\boldsymbol{\mu})$. Nonlinear approximation of shocks is a relatively new development in the field of model reduction, and hence we refer to [19, 53, 54] and references therein for specific implementations. The nonlinear model reduction approach has been applied to transonic Euler over an airfoil [19] and supersonic forward step [53].

6.4.5 Large-scale applications

We conclude this section with a few examples of model reduction applied to large-, industry-scale nonlinear aerodynamics problems.

- *Unsteady turbulent flow past Amhed body* [22]. In this work Carlberg et al. consider model reduction of nonparameterized turbulent flow over the Ahmed body modeled by detached eddy simulation. The FOM consists of $N_h \approx 1.7 \times 10^7$ spatial degrees of freedom. The FOM is hyperreduced using the GNAT method; the resulting ROM uses a reduced basis of the size $N = 283$ for the state, a reduced basis of the size $N_R = N_J = 1{,}514$ for the residual and Jacobian, and $\tilde{N}_e = 378$ sample nodes. The ROM reproduces the unsteady drag time history with less than 1 % discrepancy. The FOM requires 13 hours using 512 cores, whereas the ROM requires 3.9 hours using 4 cores; the ROM reduces the computational cost by a factor of 438.
- *Parametric shape deformation of the NASA Common Research Model* [69]. In this work Washabaugh et al. consider model reduction of steady RANS-SA flow over the NASA Common Research Model under parametric shape deformation. The FOM consists of $N_h \approx 6.8 \times 10^7$ degrees of freedom and is parameterized by four

shape parameters: wingspan, washout, streamwise wingtip rake, and vertical wingtip rake. The ROM based on the minimum-residual formulation with the gappy POD collocation hyperreduction uses $N = 23$ modes and $\tilde{N}_e = 5000$ sample nodes. The ROM achieves less than 0.3 % error in drag for test parameters considered. A single simulation of the FOM requires 2 hours using 1024 cores, whereas the ROM requires 2.8 minutes on a laptop.

6.5 Summary and conclusions

In this chapter, we surveyed model reduction techniques for linearized and nonlinear aerodynamics problems that have been developed in the past two decades. We discussed essential ingredients of model reduction, with an emphasis on techniques that are designed to address challenges in aerodynamics, including convection dominance, nonlinearity, limited stability, limited regularity, and a wide range of scales. We also reviewed successful applications of model reduction to large-scale industry-relevant aerodynamics problems to date. There still exist many open challenges to model reduction of complex aerodynamics problems. Their industrial relevance and challenging nature make them arguably an ideal testbed to develop and assess the next generation of model reduction algorithms.

Bibliography

[1] R. Abgrall and R. Crisovan, Model reduction using L1-norm minimization as an application to nonlinear hyperbolic problems, *Int. J. Numer. Methods Fluids*, **87** (12) (2018), 628–651.
[2] D. Amsallem, J. Cortial, and C. Farhat, Toward real-time computational-fluid-dynamics-based aeroelastic computations using a database of reduced-order information, *AIAA J.*, **48**(9) (2010), 2029–2037.
[3] D. Amsallem, S. Deolalikar, F. Gurrola, and C. Farhat, Model predictive control under coupled fluid-structure constraints using a database of reduced-order models on a tablet. *AIAA* 2013-2588, AIAA, 2013.
[4] D. Amsallem and C. Farhat, Interpolation method for adapting reduced-order models and application to aeroelasticity, *AIAA J.*, **46** (7) (2008), 1803–1813.
[5] D. Amsallem and C. Farhat, Stabilization of projection-based reduced-order models, *Int. J. Numer. Methods Eng.*, **91** (4) (2012), 358–377.
[6] J. S. R. Anttonen, P. I. King, and P. S. Beran, POD-based reduced-order models with deforming grids, *Math. Comput. Model.*, **38** (1–2) (2003), 41–62.
[7] D. N. Arnold, F. Brezzi, B. Cockburn, and L. D. Marini, Unified analysis of discontinuous Galerkin methods for elliptical problems, *SIAM J. Numer. Anal.*, **39** (5) (2002), 1749–1779.
[8] P. Astrid, S. Weiland, K. Willcox, and T. Backx, Missing point estimation in models described by proper orthogonal decomposition, *IEEE Trans. Autom. Control*, **53** (10) (2008), 2237–2251.
[9] M. F. Barone, I. Kalashnikova, D. J. Segalman, and H. K. Thornquist, Stable Galerkin reduced order models for linearized compressible flow, *J. Comput. Phys.*, **228** (6) (2009), 1932–1946.

[10] 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, *C. R. Acad. Sci. Paris, Ser. I*, **339** (2004), 667–672.
[11] T. Barth and P. Charrier, *Energy Stable Flux Formulas for the Discontinuous Galerkin Discretization of First-Order Nonlinear Conservation Laws*, NASA Technical Report NAS-01-001, NAS, 2001.
[12] T. J. Barth, Numerical methods for gasdynamic systems on unstructured meshes, in D. Kröner, M. Ohlberger, and C. Rohde (eds.), *An Introduction to Recent Developments in Theory and Numerics for Conservation Laws*, pp. 195–282, Springer-Verlag, 1999.
[13] T. J. Barth and M. G. Larson, *A Posteriori Error Estimates for Higher Order Godunov Finite Volume Methods on Unstructured Meshes*, Technical report, Complex Applications III, R. Herbin and D. Kroner (eds.), HERMES Science Publishing Ltd, 2002.
[14] R. Becker and R. Rannacher, An optimal control approach to a posteriori error estimation in finite element methods, *Acta Numer.*, **10** (2001), 1–102.
[15] A. N. Brooks and T. J. R. Hughes, Streamline upwind/Petrov–Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations, *Comput. Methods Appl. Mech. Eng.*, **32** (1–3) (1982), 199–259.
[16] T. Bui-Thanh, M. Damodaran, and K. Willcox, Proper orthogonal decomposition extensions for parametric applications in compressible aerodynamics. *AIAA* 2003-4213, AIAA, 2003.
[17] T. Bui-Thanh, K. Willcox, and O. Ghattas, Parametric reduced-order models for probabilistic analysis of unsteady aerodynamic applications, *AIAA J.*, **46** (10) (2008), 2520–2529.
[18] T. Bui-Thanh, K. Willcox, O. Ghattas, and B. van Bloemen Waanders, Goal-oriented, model-constrained optimization for reduction of large-scale systems, *J. Comput. Phys.*, **224** (2) (Jun 2007), 880–896.
[19] N. Cagniart, R. Crisovan, Y. Maday, and R. Abgrall, Model order reduction for hyperbolic problems: a new framework, *<hal-01583224>*, 2017.
[20] K. Carlberg, M. Barone, and H. Antil, Galerkin v. least-squares Petrov–Galerkin projection in nonlinear model reduction, *J. Comput. Phys.*, **330** (2017), 693–734.
[21] K. Carlberg, C. Bou-Mosleh, and C. Farhat, Efficient non-linear model reduction via a least-squares Petrov–Galerkin projection and compressive tensor approximations, *Int. J. Numer. Methods Eng.*, **86** (2) (2011), 155–181.
[22] K. Carlberg, C. Farhat, J. Cortial, and D. Amsallem, The GNAT method for nonlinear model reduction: effective implementation and application to computational fluid dynamics and turbulent flows, *J. Comput. Phys.*, **242** (2013), 623–647.
[23] S. Chaturantabut and D. C. Sorensen, Nonlinear model reduction via Discrete Empirical Interpolation, *SIAM J. Sci. Comput.*, **32** (5) (2010), 2737–2764.
[24] B. Cockburn, Discontinuous Galerkin methods, *Z. Angew. Math. Mech.*, **83** (11) (2003), 731–754.
[25] M. J. de C. Henshaw, K. J. Badcock, G. A. Vio, C. B. Allen, J. Chamberlain, I. Kaynes, G. Dimitriadis, J. E. Cooper, M. A. Woodgate, A. M. Rampurawala, D. Jones, C. Fenwick, A. L. Gaitonde, N. V. Taylor, D. S. Amor, T. A. Eccles, and C. J. Denley, Non-linear aeroelastic prediction for aircraft applications, *Prog. Aerosp. Sci.*, **43** (4–6) (2007), 65–137.
[26] E. H. Dowell and K. C. Hall, Modeling of fluid-structure interaction, *Annu. Rev. Fluid Mech.*, **33** (1) (2001), 445–490.
[27] J. L. Eftang, A. T. Patera, and E. M. Rønquist, An “*hp*” certified reduced basis method for parametrized elliptic partial differential equations, *SIAM J. Sci. Comput.*, **32** (6) (2010), 3170–3200.
[28] B. I. Epureanu, A parametric analysis of reduced order models of viscous flows in turbomachinery, *J. Fluids Struct.*, **17** (7) (2003), 971–982.

[29] R. Everson and L. Sirovich, Karhunen–Loève procedure for gappy data, *J. Opt. Soc. Am. A, Opt. Image Sci.*, **12** (8) (1995), 1657–1664.
[30] T. Franz, R. Zimmermann, S. Görtz, and N. Karcher, Interpolation-based reduced-order modelling for steady transonic flows via manifold learning, *Int. J. Comput. Fluid Dyn.*, **28** (3–4) (2014), 106–121.
[31] M. A. Grepl, Y. Maday, N. C. Nguyen, and A. T. Patera, Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations, *ESAIM: M2AN*, **41** (3) (2007), 575–605.
[32] B. Haasdonk and M. Ohlberger, Reduced basis method for finite volume approximations of parametrized linear evolution equations, *Math. Model. Numer. Anal.*, **42** (2) (2008), 277–302.
[33] K. C. Hall, Eigenanalysis of unsteady flows about airfoils, cascades, and wings, *AIAA J.*, **32** (12) (1994), 2426–2432.
[34] K. C. Hall, J. P. Thomas, and E. H. Dowell, Proper orthogonal decomposition technique for transonic unsteady aerodynamic flows, *AIAA J.*, **38** (10) (2000), 1853–1862.
[35] A. Harten, On the symmetric form of systems of conservation laws with entropy, *J. Comput. Phys.*, **49** (1) (1983), 151–164.
[36] S. Hovland, J. T. Gravdahl, and K. E. Willcox, Explicit model predictive control for large-scale systems via model reduction, *J. Guid. Control Dyn.*, **31** (4) (2008), 918–926.
[37] T. J. Hughes, L. P. Franca, and G. M. Hulbert, A new finite element formulation for computational fluid dynamics: VIII. The Galerkin/least-squares method for advective-diffusive equations, *Comput. Methods Appl. Mech. Eng.*, **73** (2) (1989), 173–189.
[38] F. T. Johnson, E. N. Tinoco, and N. J. Yu, Thirty years of development and application of CFD at Boeing Commercial Airplanes, Seattle, *Comput. Fluids*, **34** (10) (2005), 1115–1151.
[39] I. Kalashnikova and M. F. Barone, On the stability and convergence of a Galerkin reduced order model (ROM) of compressible flow with solid wall and far-field boundary treatment, *Int. J. Numer. Methods Eng.*, **83** (10) (2010), 1345–1375.
[40] I. Kalashnikova, M. F. Barone, and M. R. Brake, A stable Galerkin reduced order model for coupled fluid-structure interaction problems, *Int. J. Numer. Methods Eng.*, **95** (2) (2013), 121–144.
[41] C. T. Kelley and D. E. Keyes, Convergence analysis of pseudo-transient continuation, *SIAM J. Numer. Anal.*, **35** (2) (1998), 508–523.
[42] T. Kim, Frequency-domain Karhunen–Loeve method and its application to linear dynamic systems, *AIAA J.*, **36** (11) (1998), 2117–2123.
[43] P. LeGresley and J. Alonso, Airfoil design optimization using reduced order models based on proper orthogonal decomposition, *AIAA* 2000-2545, AIAA, 2000.
[44] P. A. LeGresley and J. J. Alonso, Investigation of non-linear projection for POD based reduced order models for aerodynamics, *AIAA* 2001–0926, AIAA, 2001.
[45] P. A. LeGresley and J. J. Alonso, Dynamic domain decomposition and error correction for reduced order models, *AIAA* 2003-250, AIAA, 2003.
[46] T. Lieu and C. Farhat, Adaptation of aeroelastic reduced-order models and application to an F-16 configuration, *AIAA J.*, **45** (6) (Jun 2007), 1244–1257.
[47] T. Lieu, C. Farhat, and M. Lesoinne, Reduced-order fluid/structure modeling of a complete aircraft configuration, *Comput. Methods Appl. Mech. Eng.*, **195** (41–43) (2006), 5730–5742.
[48] T. Lieu and M. Lesoinne, Parameter adaptation of reduced order models for three-dimensional flutter analysis, *AIAA* 2004-888, AIAA, 2004.
[49] D. J. Lucia, P. S. Beran, and W. A. Silva, Reduced-order modeling: new approaches for computational physics, *Prog. Aerosp. Sci.*, **40** (1–2) (2004), 51–117.
[50] Y. Maday, A. T. Patera, and D. V. Rovas, A blackbox reduced-basis output bound method for noncoercive linear problems, in *Nonlinear Partial Differential Equations and their Applications – Collège de France Seminar Volume XIV*, pp. 533–569, Elsevier, 2002.

[51] D. Mavriplis, D. Darmofal, D. Keyes, and M. Turner, Petaflops opportunities for the NASA fundamental aeronautics program, *AIAA* 2007-4084, AIAA, 2007.

[52] B. Moore, Principal component analysis in linear systems: controllability, observability, and model reduction, *IEEE Trans. Autom. Control*, **26** (1) (1981), 17–32.

[53] N. J. Nair and M. Balajewicz, Transported snapshot model order reduction approach for parametric, steady-state fluid flows containing parameter-dependent shocks, *Int. J. Numer. Methods Eng.*, **117** (12) (2019), 1234–1262.

[54] M. Ohlberger and S. Rave, Reduced basis methods: success, limitations and future challenges, in *Proceedings of the Conference Algoritmy*, pp. 1–12, 2016.

[55] M. Romanowski, Reduced order unsteady aerodynamic and aeroelastic models using Karhunen–Loeve eigenmodes, *AIAA* 1996-3981, AIAA, 1996.

[56] C. W. Rowley, Model reduction for fluids, using balanced proper orthogonal decomposition, *Int. J. Bifurc. Chaos*, **15** (03) (2005), 997–1013.

[57] 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 – Application to transport and continuum mechanics, *Arch. Comput. Methods Eng.*, **15** (3) (2008), 229–275.

[58] D. Ryckelynck, A priori hyperreduction method: an adaptive approach, *J. Comput. Phys.*, **202** (1) (2005), 346–366.

[59] R. Schmit and M. Glauser, Improvements in low dimensional tools for flow-structure interaction problems: using global POD, *AIAA* 2004-889, AIAA, 2004.

[60] L. Sirovich, Turbulence and the dynamics of coherent structures. I. Coherent structures, *Q. Appl. Math.*, **45** (3) (1987), 561–571.

[61] J. Slotnick, A. Khodadoust, J. Alonso, D. Darmofal, W. Gropp, E. Lurie, and D. Mavriplis, *CFD Vision 2030 Study: A Path to Revolutionary Computational Aerosciences*, Nasa/cr-2014-218178, NASA, 2014.

[62] P. R. Spalart and S. R. Allmaras, A one-equation turbulence model for aerodynamics flows, *Rech. Aérosp.*, **1** (1994), 5–21.

[63] J. A. Taylor and M. N. Glauser, Towards practical flow sensing and control via POD and LSE based low-dimensional tools, *J. Fluids Eng.*, **126** (3) (2004), 337.

[64] J. P. Thomas, E. H. Dowell, and K. C. Hall, Three-dimensional transonic aeroelasticity using proper orthogonal decomposition-based reduced-order models, *J. Aircr.*, **40** (3) (2003), 544–551.

[65] E. F. Toro, *Riemann Solvers and Numerical Methods for Fluid Dynamics*, Springer, 2009.

[66] A. Vendl, H. Faßbender, S. Görtz, R. Zimmermann, and M. Mifsud, Model order reduction for steady aerodynamics of high-lift configurations, *CEAS Aeronaut. J.*, **5** (4) (2014), 487–500.

[67] K. Veroy, C. Prud'homme, D. Rovas, and A. Patera, A posteriori error bounds for reduced-basis approximation of parametrized noncoercive and nonlinear elliptic partial differential equations, *AIAA* 2003-3847, AIAA, 2003.

[68] K. Washabaugh, D. Amsallem, M. Zahr, and C. Farhat, Nonlinear model reduction for CFD problems using local reduced-order bases, *AIAA* 2012-2686, AIAA, (2012).

[69] K. Washabaugh, M. J. Zahr, and C. Farhat, On the use of discrete nonlinear reduced-order models for the prediction of steady-state flows past parametrically deformed complex geometries, *AIAA* 2016-1814, AIAA, 2016.

[70] K. M. Washabaugh, *A Scalable Model Order Reduction Framework for Steady Aerodynamic Design Applications*. PhD thesis, Stanford University, 2016.

[71] K. Willcox, Unsteady flow sensing and estimation via the gappy proper orthogonal decomposition, *Comput. Fluids*, **35** (2) (2006), 208–226.

[72] K. Willcox and A. Megretski, Fourier series for accurate, stable, reduced-order models in large-scale linear applications, *SIAM J. Sci. Comput.*, **26** (3) (2005), 944–962.

[73] K. Willcox and J. Peraire, Balanced model reduction via the proper orthogonal decomposition, *AIAA J.*, **40** (11) (2002), 2323–2330.

[74] K. Willcox, J. Peraire, and J. White, An Arnoldi approach for generation of reduced-order models for turbomachinery, *Comput. Fluids*, **31** (3) (2002), 369–389.

[75] M. Yano, Discontinuous Galerkin reduced basis empirical quadrature procedure for model reduction of parametrized nonlinear conservation laws, *Adv. Comput. Math.*, **45** (5–6) (2019), 2287–2320.

[76] M. Yano, Goal-oriented model reduction of parametrized nonlinear PDEs; application to aerodynamics, *Int. J. Numer. Methods Eng.*, accepted, 2020.

[77] M. Yano and A. T. Patera, An LP empirical quadrature procedure for reduced basis treatment of parametrized nonlinear PDEs, *Comput. Methods Appl. Mech. Eng.*, **344** (2019), 1104–1123.

[78] M. J. Zahr and C. Farhat, Progressive construction of a parametric reduced-order model for PDE-constrained optimization, *Int. J. Numer. Methods Eng.*, **102** (5) (2015), 1111–1135.

[79] R. Zimmermann and S. Görtz, Improved extrapolation of steady turbulent aerodynamics using a non-linear POD-based reduced order model, *Aeronaut. J.*, **116** (1184) (2012), 1079–1100.

[80] R. Zimmermann and S. Görtz, Non-linear reduced order models for steady aerodynamics, *Proc. Comput. Sci.*, **1** (1) (2012), 165–174, ICCS 2010.