Adaptive Discontinuous-Galerkin Reduced-Basis Reduced-Quadrature Method for Many-Query CFD Problems

Eugene Du*, Michael Sleeman[†], and Masayuki Yano[‡]
Institute for Aerospace Studies, University of Toronto, Toronto, ON M3H 5T6, Canada

We present a projection-based model reduction method for efficient solution of computational fluid dynamics problems in many-query scenarios, which require the evaluation of quantities of interest for many different flow-condition, geometry, or model parameters. Our goal is to construct reduced models that provide rapid and accurate output predictions and the associated *a posteriori* error estimates. To achieve this goal, our framework builds on the following key ingredients of adaptive high-order methods: the discontinous Galerkin method, which provides stability for conservation laws; the dual-weighted residual method, which provides effective output *a posteriori* error estimates. In addition, we incorporate two model reduction ingredients: reduced bases, which provide low-dimensional empirical approximation spaces tailored for the specific parametrized problem; reduced quadrature rules, which are the tailored quadrature rules for the reduced bases constructed using an empirical quadrature procedure. Both reduced bases and reduced quadrature rules are identified through an efficient and automatic offline training procedure that is informed by the behavior of *a posteriori* error estimates. We demonstrate the efficacy and versatility of the model reduction approach in four aerodynamics problems: Reynolds-averaged Navier-Stokes (RANS) flow over the ONERA M6 wing with the Mach number and the angle of attack as the parameters; laminar flow over shape-parametrized airfoils; uncertainty quantification of RANS flow with variabilities in the empirical parameters of the Spalart-Allmaras turbulence model; and unsteady flow past NACA0012 with the Reynolds number as the parameter. The reduced models achieve ~300–20000 speedup at less than 1% drag error level relative to an adaptive DG method and provide effective error estimates.

I. Introduction

Many engineering scenarios require the evaluation of quantities of interest (i.e., output) — such as lift and drag — for different configurations (i.e., input) — such as flow conditions and geometries. Our interest is in *many-query scenarios*, which require the input-output evaluation for a large number of parameter values. Examples of many-query scenarios in aerodynamics include flight parameter sweep, design optimization, and uncertainty quantification. One approach to address many-query scenarios is to consider model reduction based on offline-online computational decomposition. In the offline stage, we train a reduced model through the exploration of the parameter space; the offline stage is expensive but is performed once. In the online stage, we invoke the reduced model to rapidly evaluate the output for many different configurations. Our emphasis in this work is the application of such a model reduction framework to computational fluid dynamics (CFD) problems in aerodynamics.

Over the past two decades, significant progress has been made in the area of projection-based model reduction for parametrized partial differential equations (PDEs) [1–3]. (We note there are non-projection-based approaches to model reduction [4, 5]; however, the present work focuses on *projection-based* model reduction.) For simple "textbook" linear elliptic PDEs, the technique is mature and can routinely reduce the computational cost by several orders of magnitude while ensuring the accuracy using reliable error bounds [1, 3]. However, there still remains a number of challenges to enable systematic model reduction of aerodynamic CFD problems, which exhibit strong nonlinearity, convection dominance, limited regularity, and a wide range of scales [2, 6].

To address these challenges, our work appeals to two of the key ingredients of adaptive high-order CFD methods. The first is the discontinuous Galerkin (DG) method [7–9], which provides stability for conservation laws and flexibility

*Current address: Computer Modeling Group, 3710 33 Street NW, Calgary, Alberta, T2L 2M1, Canada

[†] Current address: California Institute of Technology, Department of Mechanical Engineering, 1200 E California Blvd. MC 104-44, Pasadena, CA 91125, United States, msleeman@caltech.edu

[‡] Assistant Professor, University of Toronto, Institute for Aerospace Studies, 4925 Dufferin St, Toronto, ON M3H 5T6, Canada, myano@utias.utoronto.ca

for *hp* refinement. The second is the dual weighted residual (DWR) method [10, 11], which provides *a posteriori* error estimates for quantities of interest. These two methods have been successfully combined to provide efficient and reliable solution of many complex CFD problems [12]. However, while suited for single- or few-query analysis, these methods are not specifically tailored for many-query scenarios, which require CFD analysis for hundreds or thousands of different configurations.

To enable efficient *many-query* analysis, we introduce two additional ingredients of projection-based model reduction: *reduced bases* and *reduced quadrature*. A reduced basis is a set of empirical functions that is specifically tailored to approximate the parametric manifold, the set of solutions encountered as the input parameters are varied. Unlike the piecewise polynomials used in standard DG methods, the reduced basis have a global support, are specifically designed for the parametric manifold, and are constructed by solving the DG problem for a few judiciously chosen parameter values in the offline training stage [2, 3, 13]. Because the basis are tailored for the parametric manifold, we can typically provide an accurate approximation (e.g., 1% error) using much fewer reduced basis functions (∼10–100) than DG basis functions (∼10^4–10^7).

Reduced quadrature rules are quadrature rules tailored for reduced bases. The key observation is that, because there are much fewer reduced basis functions, it should be possible to significantly reduce the number of quadrature points in the DG method without compromising the accuracy of the residual evaluation. There exists a number of model reduction approaches for nonlinear PDEs that appeal to this observation, including the optimal cubature procedure [14] and energy conserving sampling and weighting (ECSW) [15, 16]. In this work, we construct these reduced quadrature rules using the *empirical quadrature procedure* (EQP) which has been devised for (continuous) finite element methods [17, 18] and for element-wise interpretation of DG methods [19, 20]. The present work extends the procedure to quadrature-point-wise interpretation of DG methods. Unlike Gauss-like quadrature rules used in standard DG methods, the empirical quadrature rules are sparse, are specifically designed for the reduced basis, and are constructed by solving accuracy-constrained sparse quadrature optimization problems in the offline training stage. Because the rule is tailored for the reduced basis, we can typically provide accurate integration with much fewer quadrature points (∼10–1000) than DG quadrature points (∼10^4–10^7).

In our reduced-basis reduced-quadrature (RB-RQ) model reduction framework, both reduced bases and reduced quadrature rules are identified in the offline training stage. We specifically identify appropriate bases and quadrature rules for both the flow problem and the adjoint problem that arise in the DWR error estimate, so that we can provide not only rapid output predictions but also *a posteriori* error estimates. As the number of basis functions and quadrature points are reduced by several orders of magnitude, the reduced models reduce the computational cost for output predictions and error estimates by several orders of magnitude.

The previous works on *projection-based* model reduction of parametrized (nonlinear) aerodynamics problems include the works by LeGresley and Alonso [21–23], Washabaugh et al [24], and Zimmermann and Görtz [25, 26]; see also review papers [27, 28]. We also refer to [4] for earlier works on *linearized* aerodynamics problems. The key differentiator of the current work is that our reduced models are equipped with online efficient *a posteriori* error estimates, which allows us to construct reduced models that meet the user-prescribed output error tolerance over the parameter range by judiciously controlling various sources of error in the model reduction process in an automated manner.

The contributions of this work are twofold. First, we develop a quadrature-point-based DG empirical quadrature procedure; the formulation provides considerable speedup relative to the element-based DG RB-EQP formulation developed in [19]. Second, we demonstrate the efficacy and versatility of the framework in four aerodynamics problems: Reynolds-averaged Navier-Stokes (RANS) flow over the ONERA M6 wing with the Mach number and the angle of attack as the parameters; laminar flow over shape-parametrized airfoils; uncertainty quantification of RANS flow with variabilities in the empirical parameters of the Spalart-Allmaras turbulence model; and unsteady flow past NACA0012 with the Reynolds number as the parameter. In the offline stage, a reduced model is trained in a fully automated manner for each problem. In the online stage, the reduced models achieve ∼300–20000 speedup at 1% drag error level relative to adaptive DG method and provide effective error estimates.

II. Formulation

A. Problem statement

We introduce a mathematical form of the parametrized CFD problems considered throughout this work. We first introduce a P-dimensional parameter domain $\mathcal{D} \subset \mathbb{R}^P$ and a d-dimensional spatial domain $\Omega \subset \mathbb{R}^d$. The system of m

conservation laws that we consider throughout this work is of the form

$$\partial_t F_0(u;\mu) + \nabla \cdot F(u, \nabla u; \mu) = S(u, \nabla u; \mu) \quad \text{in } \Omega,$$

where $F_0 : \mathbb{R}^m \times \mathcal{D} \to \mathbb{R}^m$ is the temporal flux, $F : \mathbb{R}^m \times \mathbb{R}^{d\times m} \times \mathcal{D} \to \mathbb{R}^{d\times m}$ is the spatial flux, and $S : \mathbb{R}^m \times \mathbb{R}^{d\times m} \times \mathcal{D} \to \mathbb{R}^m$ is the (in general gradient-dependent) source function. Each of these terms may depend on the parameter $\mu \in \mathcal{D}$. The conservation law is augmented with appropriate (parametrized) boundary conditions to yield a boundary value problem. Given the solution, we evaluate the associated functional output

$$s(\mu) = J(u(\mu); \mu),$$

where J is some output functional. Our goal is to construct projection-based reduced models which provide efficient approximation of the input-output map $\mathcal{D} \ni \mu \mapsto s(\mu) \in \mathbb{R}$ and equip the output with an *a posteriori* error estimate.

B. Discontinuous Galerkin method

We now review the elements of DG methods that are necessary to describe our model-reduction approach. In particular, as the approach can be illustrated more concisely without loss of generality for steady hyperbolic systems (i.e., without the unsteady, viscous, or source terms), we restrict our presentation to steady hyperbolic systems for the rest of Section II. The extensions of the framework to other classes of problems is discussed in Section III. For a more complete presentation of DG methods, we refer to [7, 9, 29].

To begin, we introduce a triangulation $\mathcal{T}_h$ of the domain Ω. We denote the set of interior and boundary facets of $\mathcal{T}_h$ by Γ^I and Γ^B, respectively; i.e., $\Gamma^I \equiv \bigcup_{\kappa\in\mathcal{T}_h} \partial\kappa \setminus \partial\Omega$ and $\Gamma^B \equiv \bigcup_{\kappa\in\mathcal{T}_h} (\partial\kappa \cap \partial\Omega)$. We then introduce a DG approximation space

$$\mathcal{V}_h \equiv \{v_h \in L^2(\Omega)^m \mid v_h|_\kappa \in \mathbb{P}^p(\kappa),\ \forall \kappa \in \mathcal{T}_h\},$$

where $L^2(\Omega)$ is the space of square integrable functions over Ω, and $\mathbb{P}^p(\kappa)$ is the space of polynomials of degree at most p over κ. The functions in the space $\mathcal{V}_h$ are in general discontinuous. We denote the dimension of the DG approximation space by $N_h \equiv \dim(\mathcal{V}_h)$.

We next introduce the (parametrized) DG residual form $r_h^{\text{exact}} : \mathcal{V}_h \times \mathcal{V}_h \times \mathcal{D} \to \mathbb{R}$ such that

$$r_h^{\text{exact}}(w_h, v_h; \mu) \equiv -\int_\Omega \nabla v_h : F(w_h;\mu)dx + \int_{\Gamma_I} (v_h^+ - v_h^-)\cdot \hat{F}(w_h^+, w_h^-, \hat{n}; \mu)ds + \int_{\Gamma_B} v_h^+ \cdot \hat{F}_B(w_h^+, \hat{n}; \mu)ds, \tag{1}$$

where $F : \mathbb{R}^m \times \mathcal{D} \to \mathbb{R}^{d\times m}$ is the spatial flux, $\hat{F} : \mathbb{R}^m \times \mathbb{R}^m \times \mathbb{R}^d \times \mathcal{D} \to \mathbb{R}^m$ is the interior numerical flux, and $\hat{F}_B : \mathbb{R}^m \times \mathbb{R}^d \times \mathcal{D} \to \mathbb{R}^m$ is the boundary numerical flux that incorporates appropriate boundary conditions. Following the convention in DG literature, the notation v_h^+ and v_h^- in the facet integrals denote the evaluation of the function $v_h \in \mathcal{V}_h$ on the + and − sides of the facet, respectively. In this work, we use Roe's approximate Riemann solver [30] for the numerical flux. Similarly, the DG output functional $J_h^{\text{exact}} : \mathcal{V}_h \times \mathcal{D} \to \mathbb{R}$ is given by

$$J_h^{\text{exact}}(w_h;\mu) \equiv \int_\Omega j^\Omega(w_h;\mu)dx + \int_{\Gamma_B} j^{\Gamma_B}(w_h^+, \nabla w_h^+; \mu)dx, \tag{2}$$

where $j^\Omega : \mathbb{R}^m \times \mathcal{D} \to \mathbb{R}$ is the domain output function, and $j^{\Gamma_B} : \mathbb{R}^m \times \mathbb{R}^{d\times m} \times \mathcal{D} \to \mathbb{R}$ is the boundary output function.

In practice, we approximate the semi-linear form (1) using a quadrature rule. To this end, we introduce (Gauss-like) quadrature rules $(x_q^\Omega, \rho_q^\Omega)_{q=1}^{Q_\Omega}$, $(x_q^{\Gamma_I}, \rho_q^{\Gamma_I})_{q=1}^{Q_{\Gamma_I}}$, and $(x_q^{\Gamma_B}, \rho_q^{\Gamma_B})_{q=1}^{Q_{\Gamma_B}}$ for the domain, interior facet, and boundary facet integration, respectively. The application of the quadrature rules to the semi-linear form (1) yields

$$r_h(w_h, v_h;\mu) \equiv \sum_{q=1}^{Q_\Omega} \rho_q^\Omega [-\nabla v_h : F(w_h)]_{x_q^\Omega} + \sum_{q=1}^{Q_{\Gamma_I}} \rho_q^{\Gamma_I} [(v_h^+ - v_h^-)\cdot \hat{F}(u_h^+, u_h^-; \hat{n})]_{x_q^{\Gamma_I}} + \sum_{q=1}^{Q_{\Gamma_B}} \rho_q^{\Gamma_B} [v_h^+ \cdot \hat{F}_B(u_h^+; \hat{n})]_{x_q^{\Gamma_B}}. \tag{3}$$

Similarly, the approximation of the quadrature rules to the output functional (2) yields

$$J_h(w_h;\mu) \equiv \sum_{q=1}^{Q_\Omega} \rho_q^\Omega [j^\Omega(w_h;\mu)]_{x_q^\Omega} + \sum_{q=1}^{Q_{\Gamma_B}} \rho_q^{\Gamma_B} [j^{\Gamma_B}(w_h^+, \nabla w_h^+; \mu)]_{x_q^{\Gamma_B}}. \tag{4}$$

In this work, we choose the domain and facet quadrature rules that exactly integrates polynomials of degree of at least $4p$.

The DG approximation of the conservation laws is as follows: given $\mu \in \mathcal{D}$, find $u_h(\mu) \in \mathcal{V}_h$ such that

$$r_h(u_h(\mu), v_h; \mu) = 0 \quad \forall v_h \in \mathcal{V}_h, \tag{5}$$

and then evaluate the output

$$s_h(\mu) \equiv J_h(u_h(\mu); \mu).$$

Because $\dim(\mathcal{V}_h) = N_h$, the weak statement (5) yields a system of N_h nonlinear equations with N_h unknowns. In this work we solve the system (5) using Newton's method with pseudo-time continuation and line search [31], where the linear system is solved using GMRES [32] preconditioned with an block-ILU(0) factorization with minimum discarded fill ordering [33]. The computational cost of DG approximation depends on two factors:

1) The first is the dimension of the approximation space $N_h = \dim(\mathcal{V}_h)$. The solution of the DG system using a Newton-like method requires multiple solutions of linear systems of the size N_h. The computational cost is hence $\mathcal{O}(N_h^\bullet)$, for some exponent greater than or equal to 1.
2) The second is the number of quadrature points $Q_h \equiv Q_\Omega + Q_{\Gamma_I} + Q_{\Gamma_B}$. The evaluation of the DG residual (3) and the associated Jacobian requires the evaluation of the fluxes and basis functions at all quadrature points. The residual evaluation cost is hence $\mathcal{O}(Q_h)$.

In the next two sections, we present a projection-based model reduction technique to significantly reduce N_h and Q_h.

C. Reduction of approximation space: reduced basis (RB) spaces

We now discuss the construction of reduced approximation spaces for parametrized problems. In particular, we construct a N-dimensional *reduced basis* space $\mathcal{V}_N \subset \mathcal{V}_h$ that is specifically tailored to approximate the parametric manifold $\{u_N(\mu)\}_{\mu \in \mathcal{D}}$ for $N \ll N_h$. To this end, we introduce a RB training parameter set $\Xi^{\text{rb}} \equiv \{\mu^1, \dots, \mu^N\} \subset \mathcal{D}$, which comprises N judiciously chosen training parameters for the reduced basis; we detail the systematic construction of Ξ^{rb} in Section II.F, and assume it is given for now. The associated RB space is

$$\mathcal{V}_N = \text{span}\{u_h(\mu)\}_{\mu \in \Xi^{\text{rb}}}.$$

By construction $\mathcal{V}_N \subset \mathcal{V}_h$ and $\dim(\mathcal{V}_N) = N$. For convenience, we also introduce the reduced basis $\{\phi_i \in \mathcal{V}_h\}_{i=1}^N$ such that $\text{span}(\{\phi_i\}_{i=1}^N) = \mathcal{V}_N$. The reduced basis allows us associate any function $v_N \in \mathcal{V}_N$ with a generalized coordinate $\mathbf{v}_N \in \mathbb{R}^N$ such that $v_N = \sum_{i=1}^N \mathbf{v}_{N,i} \phi_i$. An example of a reduced basis (for the problem considered in Section IV.B) is shown in Figure 1. The hierarchical basis captures the dominant features that the parametrized problem exhibit as the parameter is varied.

Given a RB space $\mathcal{V}_N$, we now define the associated RB problem: given $\mu \in \mathcal{D}$, find the RB solution $u_N(\mu) \in \mathcal{V}_N$ such that

$$r_h(u_N(\mu), v_N; \mu) = 0 \quad \forall v_N \in \mathcal{V}_N, \tag{6}$$

and evaluate the RB output

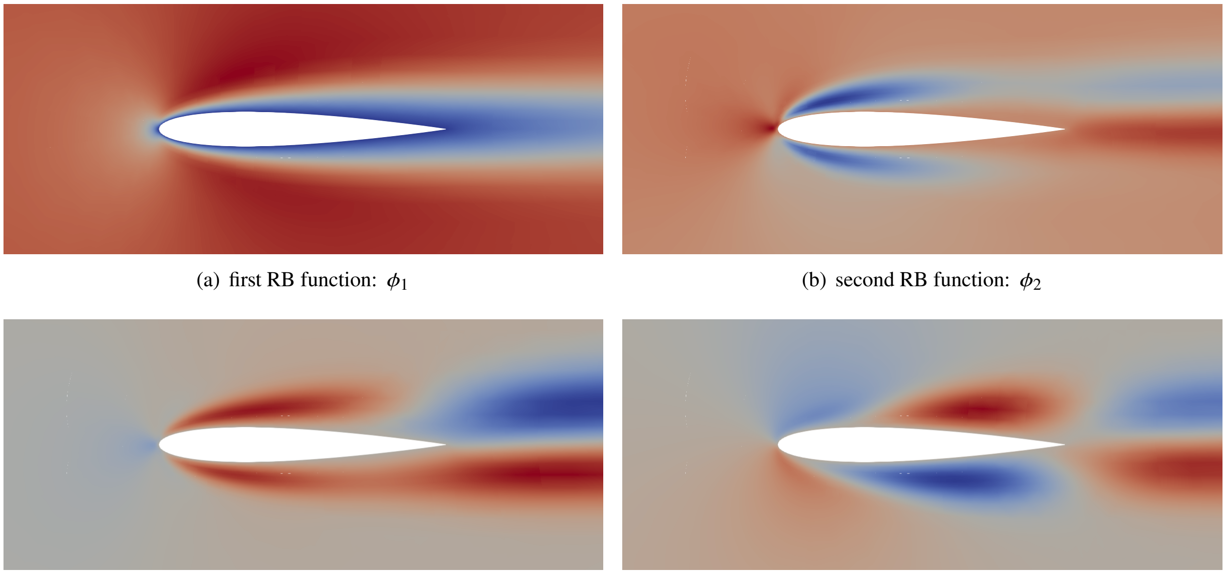
$$s_N(\mu) = J_h(u_N(\mu); \mu).$$

Because $\dim(\mathcal{V}_N) = N$, the RB problem (6) is a system of N nonlinear equations with N unknowns, where $N \ll N_h$. The system is solved using a Newton-like method. While the dimensionality of the problem has been reduced, the solution of RB problem (6) is still expensive because the evaluation of the residual (3) (and the Jacobian for a Newton-like method) scales with the number of quadrature points $Q_h = \mathcal{O}(N_h)$ (and not $\mathcal{O}(N)$).

D. Reduction of quadrature rule: reduced quadrature (RQ) using empirical quadrature procedure (EQP)

We now discuss the construction of RQ rules, which allow us to approximate the residual (3) and Jacobian in $\mathcal{O}(N)$ operations, using *the empirical quadrature procedure* [17, 18]. Specifically, our goal is to identify a sparse quadrature rule $(\tilde{x}_q^\Omega, \tilde{\rho}_q^\Omega)_{q=1}^{\tilde{Q}_\Omega}$, $(\tilde{x}_q^{\Gamma_I}, \tilde{\rho}_q^{\Gamma_I})_{q=1}^{\tilde{Q}_{\Gamma_I}}$, and $(\tilde{x}_q^{\Gamma_B}, \tilde{\rho}_q^{\Gamma_B})_{q=1}^{\tilde{Q}_{\Gamma_B}}$ for the domain, interior facet, and boundary facet integration so that $\tilde{Q}_\Omega \ll Q_\Omega$, $\tilde{Q}_{\Gamma_I} \ll Q_{\Gamma_I}$, and $\tilde{Q}_{\Gamma_B} \ll Q_{\Gamma_B}$, and then to construct a RQ semi-linear form

$$\tilde{r}_h(w_h, v_h; \mu) \equiv \sum_{q=1}^{\tilde{Q}_\Omega} \tilde{\rho}_q^\Omega [-\nabla v_h : F(w_h)]_{\tilde{x}_q^\Omega} + \sum_{q=1}^{\tilde{Q}_{\Gamma_I}} \tilde{\rho}_q^{\Gamma_I} [(v_h^+ - v_h^-) \cdot \hat{F}(u_h^+, u_h^-; \hat{n})]_{\tilde{x}_q^{\Gamma_I}} + \sum_{q=1}^{\tilde{Q}_{\Gamma_B}} \tilde{\rho}_q^{\Gamma_B} [v_h^+ \cdot \hat{F}_B(u_h^+; \hat{n})]_{\tilde{x}_q^{\Gamma_B}}. \tag{7}$$



(a) first RB function: ϕ_1

(b) second RB function: ϕ_2

(c) third RB function: ϕ_3

(d) fourth RB function: ϕ_4

Fig. 1 An example of a reduced basis: the first four basis functions for the parametrized shape transformation problem considered in Section IV.B.

The total number of quadrature points is $\tilde{Q} \equiv \tilde{Q}_\Omega + \tilde{Q}_{\Gamma_I} + \tilde{Q}_{\Gamma_B}$.

We wish the RQ rules to satisfy two properties. First, we wish the rules to be sparse so that $\tilde{Q} \ll Q_h$; this condition ensures rapid evaluation of the residual. Second, we wish the rules to be accurate so that $\tilde{r}_h(\cdot,\cdot;\mu) \approx r_h(\cdot,\cdot;\mu)$; this condition ensures that the solution associated with the RQ rule is accurate. Specifically, we wish the error in the output (i.e., the quantity of interest) due to the use of the RQ rule to be less than some user-prescribed tolerance δ^{eqp}. We cast this problem of finding a sparse quadrature rule that satisfies the output accuracy constraint as a constrained optimization problem.

We now deduce appropriate accuracy constraints for the optimization problem. To this end, we denote by $\tilde{u}_N(\mu) \in \mathcal{V}_N$ the RB-RQ solution associated with the RQ rules; i.e., $\tilde{u}_N(\mu) \in \mathcal{V}_N$ satisfies $\tilde{r}_h(\tilde{u}_N(\mu), v_N;\mu) = 0$ $\forall v_N \in \mathcal{V}_N$. We appeal to the dual-weighted residual error relationship [10, 20] to obtain

$$|J_h(u_N(\mu);\mu) - J_h(\tilde{u}_N(\mu);\mu)| \approx |r_h(u_N(\mu), z_N(\mu);\mu) - \tilde{r}_h(u_N(\mu), z_N(\mu);\mu)|, \tag{8}$$

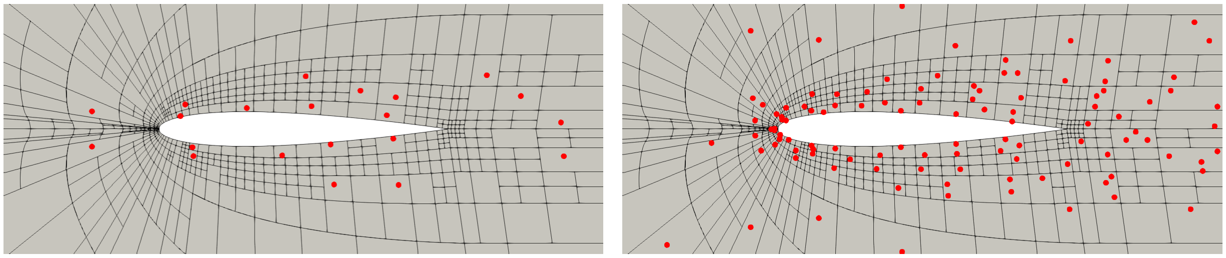
where the dual solution $z_N(\mu) \in \mathcal{V}_N$ satisfies

$$r_h'(u_N(\mu); w_N(\mu), z_N(\mu)) = J_h'(u_N(\mu); w_N(\mu)) \quad \forall z_N \in \mathcal{V}_N.$$

Hence, in order to control the error in the output due to the use of the reduced quadrature, we must control the error in the dual-weighted residual (8).

We now solve the RQ optimization problem using the EQP [17, 18]. We first introduce a training parameter set $\Xi^{\text{eqp}} \equiv \{\hat{\mu}^1, \dots, \hat{\mu}^{N_{\text{eqp}}}\} \subset \mathcal{D}$. We denote the associate training state set by $U^{\text{eqp}} \equiv \{\hat{u}_N^1, \dots, \hat{u}_N^{N_{\text{eqp}}}\}$ and the training dual state set by $Z^{\text{eqp}} \equiv \{\hat{z}_N^1, \dots, \hat{z}_N^{N_{\text{eqp}}}\}$. Our constrained problem optimization is as follows: find sparse and non-negative quadrature weights

$$(\hat{\rho}^{\Omega,\star}, \hat{\rho}^{\Gamma_I,\star}, \hat{\rho}^{\Gamma_B,\star}) = \underset{(\hat{\rho}^{\Omega},\hat{\rho}^{\Gamma_I},\hat{\rho}^{\Gamma_B}) \in \mathbb{R}^{Q_\Omega}_{\geq 0} \times \mathbb{R}^{Q_{\Gamma_I}}_{\geq 0} \times \mathbb{R}^{Q_{\Gamma_B}}_{\geq 0}}{\arg\min} (\|\hat{\rho}^{\Omega}\|_0 + \|\hat{\rho}^{\Gamma_I}\|_0 + \|\hat{\rho}^{\Gamma_B}\|_0) \tag{9}$$



(a) empirical quadrature points for $\{\phi_i^{\rm pr}\}_{i=1}^{3}$

(b) empirical quadrature points for $\{\phi_i^{\rm pr}\}_{i=1}^{13}$

Fig. 2 Examples of empirical quadrature points: empirical quadrature points associated with two different reduced bases of the parametrized shape transformation problem considered in Section IV.B.

such that the accuracy constraints (8) are satisfied at all $N_{\rm eqp}$ training points in the sense that

$$\Bigg|\sum_{q=1}^{Q_\Omega} \hat\rho_q^\Omega [-\nabla \hat z_N^j : F(\hat u_N^j;\hat\mu^j)]_{x_q^\Omega} + \sum_{q=1}^{Q_{\Gamma_I}} \hat\rho_q^{\Gamma_I} [(\hat z_N^{j,+} - \hat z_N^{j,-})\cdot \hat F(\hat u_N^{j,+}, \hat u_N^{j,-}, \hat n; \hat\mu^j)]_{x_q^{\Gamma_I}}$$

$$+ \sum_{q=1}^{Q_{\Gamma_B}} \hat\rho_q^{\Gamma_B}[\hat z_N^{j,+}\cdot \hat F_B(\hat u_N^{j,+},\hat n;\hat\mu^j)]_{x_q^{\Gamma_B}} - r_h(\hat u_N^j, \hat z_N^j;\hat\mu^j)\Bigg| \le \delta^{\rm eqp}, \quad j=1,\dots,N_{\rm eqp}. \tag{10}$$

Here $\|\cdot\|_0$ denotes the 0-norm, which counts the number of nonzero entries. The sum of the first three terms of the constraints yields the "approximated" residual $r_h(\hat u^j, \hat z^j; \hat\mu^j)$ evaluated using the quadrature weights $\hat\rho^\Omega$, $\hat\rho^{\Gamma_I}$, and $\hat\rho^{\Gamma_B}$ to be optimized, and the last term of the constraint is the "truth" residual $r_h(\hat u_N^j, \hat z_N^j; \hat\mu^j)$; the constraint hence enforces the difference between the "approximated" residual and the "truth" residual to be smaller than the user-prescribed tolerance $\delta^{\rm eqp}$ for all $N_{\rm eqp}$ parameter values in the EQP training set $\Xi^{\rm eqp}$. In addition, the constraints are linear in the quadrature weights $\hat\rho^\Omega$, $\hat\rho^{\Gamma_I}$, and $\hat\rho^{\Gamma_B}$, which simplifies the solution of the optimization problem. Detailed discussions and theoretical analyses of the EQP is provided in [18–20].

We solve the optimization problem (9) approximately as a non-negative least-squares (NNLS) problem using a parallel NNLS algorithm developed for the ECSW method [15, 16]. We then set the RQ rules $(\tilde x_q^\Omega, \tilde\rho_q^\Omega)_{q=1}^{\tilde Q_\Omega}$, $(\tilde x_q^{\Gamma_I}, \tilde\rho_q^{\Gamma_I})_{q=1}^{\tilde Q_{\Gamma_I}}$, and $(\tilde x_q^{\Gamma_B}, \tilde\rho_q^{\Gamma_B})_{q=1}^{\tilde Q_{\Gamma_B}}$ in (7) by collecting nonzero entries of the sparse weights $\hat\rho^{\Omega,\star}$, $\hat\rho^{\Gamma_I,\star}$, and $\hat\rho^{\Gamma_B,\star}$, respectively.

Figure 2 shows examples of RQ rules (for the problem considered in Section IV.B). We observe that the quadrature points are sparse; as we will see in Section IV.B, the reduced quadrature rule contains ≈ 1.5% of the original "truth" quadrature points, yet achieves less than 1% drag error for all parameter values considered.

We reduce the quadrature rule associated with the output functional (4) in a similar manner. Namely, we consider the constrained optimization problem (9), but replace the residual constraints (10) with the output-functional constraints

$$\Bigg|\sum_{q=1}^{Q_\Omega} \hat\rho_q^\Omega [j^\Omega(\hat u_h^j;\hat\mu^j)]_{x_q^\Omega} + \sum_{q=1}^{Q_{\Gamma_B}} \hat\rho_q^{\Gamma_B}[j^{\Gamma_B}(\hat u_h^{j+}, \nabla\hat u_h^{j+};\hat\mu^j)]_{x_q^{\Gamma_B}} - J_h(\hat u_h^j;\hat\mu^j)\Bigg| \le \delta^{\rm eqp}, \quad j=1,\dots,N_{\rm eqp}. \tag{11}$$

The constraint is again linear in the quadrature weights $\hat\rho^\Omega$ and $\hat\rho^{\Gamma_B}$, and the optimization problem is solved using the parallel NNLS algorithm in [16].

Having found the RQ rules, our RB-RQ problem is as follows: given $\mu \in \mathcal{D}$, find the RB-RQ solution $\tilde u_N(\mu) \in \mathcal{V}_N$ such that

$$\tilde r_h(\tilde u_N(\mu), v_N; \mu) = 0 \quad \forall v_N \in \mathcal{V}_N, \tag{12}$$

and then evaluate the RB-RQ output

$$\tilde s_N(\mu) = J_h(\tilde u_N(\mu); \mu). \tag{13}$$

Give that the number of degrees of freedom is $\dim(V_N) = N \ll N_h$ and the number of quadrature points is $\tilde Q \ll Q_h$ (and in particular of $\tilde Q = O(N)$), this RB-RQ problem can be solved for any parameter value $\mu \in \mathcal{D}$ in $O(N^\bullet)$ computational cost.

E. *A posteriori* error estimation

We now wish to equip our RB-RQ output given by (12) and (13) with an *a posteriori* error estimate. To this end, we note that there are two distinct sources of error in our RB-RQ output: the first is the error between the true PDE solution and the underlying DG-FEM solution; the second is the error between the DG-FEM solution and the RB-RQ solution. Mathematically, we can decompose the error as

$$|s(\mu) - \tilde{s}_N(\mu)| \leq \underbrace{|s(\mu) - s_h(\mu)|}_{\text{FE error}} + \underbrace{|s_h(\mu) - \tilde{s}_N(\mu)|}_{\text{RB-RQ error}}.$$

We estimate and control the output error as follows. First, in the offline stage, we estimate and adaptively refine the snapshots (and hence the reduced basis) so that the FE error is less than the user-prescribed tolerance for all snapshot parameter values in Ξ^{rb}. (This in general does *not* guarantee the FE error meets the tolerance for *all* parameter values in $\mathcal{D}$, but to our knowledge there does not exist a method to estimate the FE error for all parameter values in $\mathcal{D}$.) Second, we estimate the RB-RQ error in an online-efficient manner (i.e., in $O(N)$ operations).

To estimate the FE error, we use the standard dual-weighted residual (DWR) method for FE discretizations [10, 12]. To this end, we first introduce an enriched DG space by increasing the polynomial degree by one: $\mathcal{V}_{\hat{h}} \equiv \{v_h \in L^2(\Omega)^m \mid v_h|_\kappa \in \mathbb{P}^{p+1}(\kappa),\ \forall \kappa \in \mathcal{T}_h\}$. We then compute the dual (or adjoint) solution in the enrich space: given $\mu \in \mathcal{D}$ and $u_h(\mu) \in \mathcal{V}_h$, find $z_{\hat{h}}(\mu) \in \mathcal{V}_{\hat{h}}$ such that

$$r_h'(u_h(\mu); w_{\hat{h}}, z_{\hat{h}}(\mu); \mu) = J_h'(u_h(\mu); w_{\hat{h}}; \mu) \quad \forall w_h \in \mathcal{V}_{\hat{h}}.$$

We then evaluate the error estimate

$$|s(\mu) - s_h(\mu)| \approx \eta_h^{\mathrm{fe}}(\mu) \equiv |r_h(u_h(\mu), z_{\hat{h}}(\mu); \mu)|.$$

The efficacy of this error estimate for CFD problems is discussed in [10]. The error estimate can also be localized to drive output-based adaptive mesh refinement, which is important to realize the full potential of high-order DG methods in aerodynamics problems; see, e.g., [29, 34].

To estimate the RB-RQ error in an online-efficient manner, we construct an RB-RQ approximation of the DWR error estimate. To this end, we first introduce an RB approximation space tailored for dual solutions:

$$\mathcal{V}_N^{\mathrm{du}} = \mathrm{span}\{z_h(\mu)\}_{\mu \in \Xi^{\mathrm{rb}}}.$$

In general $\mathcal{V}_N \neq \mathcal{V}_N^{\mathrm{du}}$ as we expect the primal and dual solutions to exhibit different behaviors, particularly for hyperbolic and advection-dominated PDEs in aerodynamics. The RB approximation of the dual problem is as follows: given $\mu \in \mathcal{D}$ and $\tilde{u}_N(\mu) \in \mathcal{V}_N$, find $z_N^{\mathrm{du}}(\mu) \in \mathcal{V}_N^{\mathrm{du}}$ such that

$$r_h'(\tilde{u}_N(\mu); w_N, z_N^{\mathrm{du}}(\mu); \mu) = J_h'(\tilde{u}_N(\mu); w_N; \mu) \quad \forall w_N \in \mathcal{V}_N^{\mathrm{du}}. \tag{14}$$

We then evaluate the error estimate

$$|s(\mu) - \tilde{s}_N(\mu)| \approx \eta_N^{\mathrm{rb}}(\mu) \equiv |r_h(\tilde{u}_N(\mu), z_N^{\mathrm{du}}(\mu); \mu)|. \tag{15}$$

This error estimate, however, is not online efficient because the evaluation of the Jacobian in (14) and the residual in (15) requires the evaluation of the DG form (3) which uses $Q_h \gg N$ "truth" quadrature points.

To enable online efficient evaluation of the error estimate, we again apply the EQP procedure to (14) and (15) to find their RQ approximations. To this end, we introduce an empirical quadrature rule $(\tilde{x}_q^{\Omega}, \tilde{\rho}_q^{\Omega})_{q=1}^{\tilde{Q}_\Omega}$, $(\tilde{x}_q^{\Gamma_I}, \tilde{\rho}_q^{\Gamma_I})_{q=1}^{\tilde{Q}_{\Gamma_I}}$, and $(\tilde{x}_q^{\Gamma_B}, \tilde{\rho}_q^{\Gamma_B})_{q=1}^{\tilde{Q}_{\Gamma_B}}$ associated with the domain, interior facet, and boundary facet for $\tilde{Q}_\Omega \ll Q_\Omega$, $\tilde{Q}_{\Gamma_I} \ll Q_{\Gamma_I}$, and $\tilde{Q}_{\Gamma_B} \ll Q_{\Gamma_B}$. We then construct an RQ approximated residual of the form (7). We tailor these constrains to the solution of the dual problem (14) and the evaluation of the DWR (15) (instead of the solution of the primal problem (12)). To this end, we first introduce the solution to the dual of the dual problem (i.e., the tangent problem): find $y_N(\mu) \in \mathcal{V}_N^{\mathrm{du}}$ such that

$$r_h'(\tilde{u}_N(\mu); y_N(\mu), v_N; \mu) = r_h(\tilde{u}_N(\mu); v_N; \mu) \quad \forall v_N \in \mathcal{V}_N^{\mathrm{du}}.$$

We then introduce a training parameter set $\Xi^{\mathrm{eqp}} = \{\hat{\mu}_1, \ldots, \hat{\mu}_{N_{\mathrm{eqp}}}\}$ and the associated training state set $U^{\mathrm{eqp}} = \{\hat{u}^j \equiv \tilde{u}(\hat{\mu}^j)\}_{j=1}^{N_{\mathrm{eqp}}}$, training dual state set $Z^{\mathrm{eqp,du}} \equiv \{\hat{z}^j \equiv z_N^{\mathrm{du}}(\hat{\mu}^j)\}_{j=1}^{N_{\mathrm{eqp}}}$ and training tangent state set $Y^{\mathrm{eqp,du}} \equiv \{\hat{y}^j \equiv y_N^{\mathrm{du}}(\hat{\mu}^j)\}_{j=1}^{N_{\mathrm{eqp}}}$. The appropriate constraints for the dual problem (14) is

$$|(r'_h(\hat{u}_N^j; \hat{y}_N^j, \hat{z}_N^j; \hat{\mu}^j) - J'_h(\hat{u}_N^j; \hat{y}_N^j; \hat{\mu}^j)) - (\tilde{r}'_h(\hat{u}_N^j; \hat{y}_N^j, \hat{z}_N^j; \hat{\mu}^j) - \tilde{J}'_h(\hat{u}_N^j; \hat{y}_N^j; \hat{\mu}^j))| \leq \delta^{\mathrm{eqp}}, \quad j = 1, \ldots, N_{\mathrm{eqp}}, \tag{16}$$

and for the DWR evaluation (15) is

$$|r_h(\hat{u}_N^j, \hat{z}_N^j; \hat{\mu}^j) - \tilde{r}_h(\hat{u}_N^j, \hat{z}_N^j; \hat{\mu}^j)| \leq \delta^{\mathrm{eqp}}, \quad j = 1, \ldots, N_{\mathrm{eqp}}. \tag{17}$$

As before, we consider the constrained optimization problem (9), but replace the residual constraints (10) with the above two constraints. Note that the terms $r'_h(\hat{u}_N^j; \hat{y}_N^j, \hat{z}_N^j; \hat{\mu}^j)$, $\tilde{q}'_h(\hat{u}_N^j; \hat{y}_N^j; \hat{\mu}^j)$, and $\tilde{r}_h(\hat{u}_N^j, \hat{z}_N^j; \hat{\mu}^j)$ are expanded in terms of quadrature weights $\hat{\rho}^{\Omega}$, $\hat{\rho}^{\Gamma_I}$, and $\hat{\rho}^{\Gamma_B}$, as it was done in (10) and (11). The resulting constraints are linear in the quadrature weights, and the optimization problem is solved using the parallel NNLS algorithm in [16]. Detailed discussions and theoretical analyses of the RB-RQ approximation of the DWR error estimates is provided in [20].

Given the reduced basis and empirical quadrature rule for the DWR error estimates, our online-efficient error estimate is given by the following: given $\mu \in \mathcal{D}$ and $\tilde{u}_N(\mu) \in \mathcal{V}_N$, find the RB-RQ dual solution $\tilde{z}_N(\mu) \in \mathcal{V}_N^{\mathrm{du}}$ such that

$$\tilde{r}'_h(\tilde{u}_N(\mu), w_N, \tilde{z}_N(\mu); \mu) = \tilde{J}'_h(\tilde{u}_N(\mu), w_N; \mu) \quad \forall w_N \in \mathcal{V}_N^{\mathrm{du}}, \tag{18}$$

and then evaluate the RB-RQ DWR error estimate

$$|s(\mu) - \tilde{s}_N(\mu)| \approx \tilde{\eta}_N^{\mathrm{rb}}(\mu) \equiv |\tilde{r}_h(\tilde{u}_N(\mu), \tilde{z}_N(\mu); \mu)|. \tag{19}$$

The cost to evaluate the error estimate is $O(N^\bullet \ll N_h^\bullet)$ and $O(\tilde{Q} \ll Q_h)$, and hence it can be evaluated in an online-efficient manner.

F. Offline training: simultaneous adaptive FE, RB, and RQ training

We now describe a greedy algorithm [1, 3], which constructs the RB-RQ reduced model in an efficient manner. The particular algorithm we use is the adaptive algorithm that simultaneously constructs the FE space, RB spaces, and RQ rules based on the behavior of the online-efficient *a posteriori* error estimate [20]. We present the algorithm in Algorithm 1 for completeness, but refer to [20] for details. The algorithm takes as its input a greedy training set $\Xi^{\mathrm{train}} \subset \mathcal{D}$ and an EQP training set $\Xi^{\mathrm{eqp}} \subset \mathcal{D}$ as well as the user-prescribed tolerances for FE space, RB space, and RQ rule. Its goal is to identify appropriate primal and dual reduced bases $\{\phi_i\}_{i=1}^{N_{\max}}$ and $\{\phi_i^{\mathrm{du}}\}_{i=1}^{N_{\max}}$, respectively, and the associated RQ rules for the residual, output, and DWR that meet the user-prescribed tolerance.

To begin, the algorithm evaluates the RB-RQ solution and the associated *a posteriori* error estimate $\tilde{\eta}_{N-1}^{\mathrm{rb}}(\mu)$ for all parameters $\mu \in \Xi^{\mathrm{train}}$ and then choose the parameter that maximizes the error estimate (i.e., the parameter that is least-well approximated by the current RB-RQ discretization) as the next sampling point μ^N (line 3). It next use the adaptive DG method to solve for the primal solution $u_h(\mu^N)$ and dual solution $z_h(\mu^N)$ that meets the user-prescribed FE error tolerance (line 5). It then augments the parameter set Ξ^{rb} (line 6) and the reduced bases (line 7); the reduced bases are orthogonalized using the Gram-Schmidt method for numerical stability. We then update the RQ rules for the residual, output, and DWR by solving the quadrature optimization problems (9) with constraints (10), (11), as well as (16) and (17), respectively (line 8). This greedy iteration is repeated until the user prescribed RB tolerance is met.

Once the reduced bases $\{\phi_i\}_{i=1}^{N_{\max}}$ and $\{\phi_i^{\mathrm{du}}\}_{i=1}^{N_{\max}}$ and quadrature rules $\left((\tilde{x}_q^{\Omega}, \tilde{\rho}_q^{\Omega})_{q=1}^{\tilde{Q}^{\Omega}}, (\tilde{x}_q^{\Gamma_I}, \tilde{\rho}_q^{\Gamma_I})_{q=1}^{\tilde{Q}_{\Gamma_I}}, (\tilde{x}_q^{\Gamma_B}, \tilde{\rho}_q^{\Gamma_B})_{q=1}^{\tilde{Q}_{\Gamma_B}}\right)_{\mathrm{res}}$ for the residual (12), $\left((\tilde{x}_q^{\Omega}, \tilde{\rho}_q^{\Omega})_{q=1}^{\tilde{Q}^{\Omega}}, (\tilde{x}_q^{\Gamma_I}, \tilde{\rho}_q^{\Gamma_I})_{q=1}^{\tilde{Q}_{\Gamma_I}}, (\tilde{x}_q^{\Gamma_B}, \tilde{\rho}_q^{\Gamma_B})_{q=1}^{\tilde{Q}_{\Gamma_B}}\right)_{\mathrm{out}}$ for the output functional (13), and $\left((\tilde{x}_q^{\Omega}, \tilde{\rho}_q^{\Omega})_{q=1}^{\tilde{Q}^{\Omega}}, (\tilde{x}_q^{\Gamma_I}, \tilde{\rho}_q^{\Gamma_I})_{q=1}^{\tilde{Q}_{\Gamma_I}}, (\tilde{x}_q^{\Gamma_B}, \tilde{\rho}_q^{\Gamma_B})_{q=1}^{\tilde{Q}_{\Gamma_B}}\right)_{\mathrm{dwr}}$ for the DWR error estimate (18) and (19) are identified, we evaluate and store the reduced bases at the quadrature points, so that they can be used to evaluate the RB-RQ residual, output functional, and DWR estimates. For instance, the residual (and Jacobian) evaluation in (12) requires the values of the basis functions at the quadrature points (i.e., $\{\{\phi_i(\tilde{x}_q^{\Omega})\}_{q=1}^{Q_\Omega}\}_{i=1}^{N_{\mathrm{ma}\tilde{x}}}$, $\{\{\phi_i(\tilde{x}_q^{\Gamma_I})\}_{q=1}^{Q_{\Gamma_I}}\}_{i=1}^{N_{\max}}$, $\{\{\phi_i(\tilde{x}_q^{\Gamma_B})\}_{q=1}^{Q_{\Gamma_B}}\}_{i=1}^{N_{\max}}$) and the gradients at the quadrature points (i.e., $\{\{\nabla\phi_i(\tilde{x}_q^{\Omega})\}_{q=1}^{Q_\Omega}\}_{i=1}^{N_{\max}}$, $\{\{\nabla\phi_i(\tilde{x}_q^{\Gamma_I})\}_{q=1}^{Q_{\Gamma_I}}\}_{i=1}^{N_{\max}}$, $\{\{\nabla\phi_i(\tilde{x}_q^{\Gamma_B})\}_{q=1}^{Q_{\Gamma_B}}\}_{i=1}^{N_{\max}}$). Our RB-RQ model stores these basis function values and the quadrature weights (i.e., $\tilde{\rho}^{\Omega}$, $\tilde{\rho}^{\Gamma_I}$, $\tilde{\rho}^{\Gamma_B}$) so that it can solve the

Algorithm 1 Greedy algorithm for offline training.

```
Input:
    training sets: $\Xi^{\text{train}}_{N_{\text{train}}} \subset \mathcal{D}$, $\Xi^{\text{eqp}}_{N_{\text{eqp}}} \subset \mathcal{D}$.
    FE, RB, and EQP tolerances: $\delta^{\text{fe}} \in \mathbb{R}_{>0}$, $\delta^{\text{rb}} \in \mathbb{R}_{>0}$, $\delta^{\text{eqp}} \in \mathbb{R}_{\geq 0}$
Output:
    reduced bases: $\{\phi_i\}_{i=1}^{N_{\max}}$, $\{\phi_i^{\text{du}}\}_{i=1}^{N_{\max}}$
    RQ rules: three sets of rules $\left((\tilde{x}_q^{\Omega}, \tilde{\rho}_q^{\Omega})_{q=1}^{\tilde{Q}^{\Omega}}, \tilde{x}_q^{\Gamma_I}, \tilde{\rho}_q^{\Gamma_I})_{q=1}^{\tilde{Q}_{\Gamma_I}}, (\tilde{x}_q^{\Gamma_B}, \tilde{\rho}_q^{\Gamma_B})_{q=1}^{\tilde{Q}_{\Gamma_B}}\right)$ for residual, output, and DWR.

Set $\Xi^{\text{rb}}_{N=0} = \emptyset$, $\{\phi_i^{\text{pr}}\}_{i=1}^{0} = \{\phi_i^{\text{du}}\}_{i=1}^{0} = \emptyset$.
for $N = 1, \dots, N_{\max}$ do
    Set $\mu^N = \arg\sup_{\mu \in \Xi^{\text{train}}_{N_{\text{train}}}} \tilde{\eta}^{\text{rb}}_{N-1}(\mu)$.
    If $\tilde{\eta}^{\text{rb}}_{N}(\mu^N) < \delta^{\text{rb}}$, terminate.
    Find DG solutions $u_h(\mu^N) \in \mathcal{V}_h$ and $z_h(\mu^N) \in \mathcal{V}_h$; adapt as necessary such that $\eta_h^{\text{fe}}(\mu^N) \leq \delta^{\text{fe}}$.
    Update parameter set: $\Xi^{\text{rb}}_N = \Xi^{\text{rb}}_{N-1} \cup \mu^N$.
    Update RB: $\{\phi_i^{\text{pr}}\}_{i=1}^{N} = \mathrm{GS}_{\mathcal{V}}\{\{\phi_i^{\text{pr}}\}_{i=1}^{N-1}, u_h(\mu^N)\}$; $\{\phi_i^{\text{du}}\}_{i=1}^{N} = \mathrm{GS}_{\mathcal{V}}\{\{\phi_i^{\text{du}}\}_{i=1}^{N-1}, z_h(\mu^N)\}$.
    Update RQ rules: solve (9) for residual, output, and DWR with appropriate accuracy constraints.
end for
```

RB-RQ problem (12) efficiently in the online stage. We similarly store (i) the primal basis evaluated at the output RQ points to provide online-efficient evaluation of the RB-RQ output (13) and (ii) the primal and dual bases evaluated at the DWR RQ points to provide online-efficient evaluation of the DWR error estimate (18) and (19).

III. Extensions

In Section II, we limited our presentation of the RB-RQ framework to steady hyperbolic equations for simplicity. We now provide an overview of extensions to the framework to treat (A) viscous and source terms, (B) unsteady problems, (C) parametrized shapes, and (D) high-dimensional problems.

A. Viscous and source terms

To extend the formulation to viscous flows, we incorporate the symmetric interior penalty (IP) discretization [29, 35]. We use the IP discretization because the facet penalty terms takes a simple form that is amenable to our RQ approximation. Other viscous discretizations, such as BR2 and CDG, require the computation of element-dependent lifting operators, which do not permit an efficient RQ approximation. The use of the IP method differs from the earlier RB-RQ method which used an element-wise (instead of the quadrature-point-wise) decomposition and the BR2 discretization [19, 20].

Similarly, in the presence of gradient-dependent source functions (e.g., in the RANS equations), we use the so-called dual-inconsistent discretization which avoids evaluation of the lifting operators [36]. We have not observed the negative impact of the dual-inconsistent treatment of the source term for RANS problems we have studied. The use of the dual-inconsistent source discretization differs from the earlier RB-RQ method that used an element-wise decomposition and an asymptotically dual-consistent source discretization [20].

B. Unsteady problems

We now discuss the extension of the RB-RQ method for unsteady problems. To extend the formulation to unsteady problems, we modify the construction of both reduced bases and reduced quadrature rules. We refer to [37, 38] for details and provide an overview of the extension.

To construct a reduced basis, we apply the so-called POD-Greedy procedure [39], which we briefly describe. Suppose we have a reduced basis space $\mathcal{V}_N$ at a given iteration of the greedy algorithm (Algorithm 1). We first identify the parameter value $\mu' \in \Xi^{\text{train}}$ that maximizes the *a posteriori* error estimate. We next solve the problem at μ' using the DG method and collect snapshots $\{u_h(t^k; \mu')\}_{k=1}^{K}$ for all K time instances. We then remove the components of the snapshots that lies in $\mathcal{V}_N$ to construct $\{\Pi^{\perp}_{\mathcal{V}_N} u_h(t^k; \mu')\}_{k=1}^{K}$. We finally apply proper orthogonal decomposition (POD) and augment the current space $\mathcal{V}_N$ with the new POD basis to obtain $\mathcal{V}_{N+\delta N}$. The procedure can be concisely written as

$$\mathcal{V}_{N+\delta N} = \mathcal{V}_N + \mathrm{POD}_{\delta N}\{\Pi_{\mathcal{V}_N^\perp} u_h(t^k;\mu')\}_{k=1}^K.$$

This procedure is repeated in the greedy algorithm.

To construct a RQ rule, we again solve the constrained optimization problem (9). However, we replace the accuracy constraints for the steady residual (8) (or more explicitly (10)) with the unsteady residual that arises in a given step of a multi-step or multi-stage method. For example, if the backward Euler method is used, the constraint is of the form

$$|r_h^{\text{unsteady}}(\hat{u}_N^j, \hat{z}_N^j; \hat{\mu}^j) - \tilde{r}_h^{\text{unsteady}}(\hat{u}_N^j, \hat{z}_N^j; \hat{\mu}^j)| \leq \delta^{\text{eqp}},$$

where the unsteady residual for the backward Euler method and the mass forms are given by

$$\begin{aligned} r_h^{\text{unsteady}}(w_N, v_N; \mu) &\equiv m(w_N(t^k), v_N(t^k); \hat{\mu}) - m(w_N(t^{k-1}), v_N(t^{k-1}); \hat{\mu}) + \Delta t r_h(w_N(t^k), v_N(t^k); \hat{\mu}^j), \\ m(w_N, v_N; \mu) &\equiv \sum_{q=1}^{M_\Omega} \rho_q^\Omega [v_N \cdot F_0(w_N; \mu)]_{x_q^\Omega} \approx \int_\Omega v_N \cdot F_0(w_N; \mu) dx, \end{aligned}$$

and $\tilde{r}_h^{\text{unsteady}}(\cdot, \cdot; \cdot, \cdot)$ and $\tilde{m}_h(\cdot, \cdot; \cdot)$ are their RQ approximations based on $(\tilde{x}_q^\Omega, \tilde{\rho}_q^\Omega)_{q=1}^{\tilde{Q}_\Omega}$, $(\tilde{x}_q^{\Gamma_I}, \tilde{\rho}_q^{\Gamma_I})_{q=1}^{\tilde{Q}_{\Gamma_I}}$, and $(\tilde{x}_q^{\Gamma_B}, \tilde{\rho}_q^{\Gamma_B})_{q=1}^{\tilde{Q}_{\Gamma_B}}$. In other words, we ensure that the RQ approximation yields an accurate *unsteady* residual.

This procedure readily extends to any multi-step or multi-stage method; in Section IV.D we apply the method to a three-stage diagonally implicit Runge-Kutta (DIRK) method [40]. In addition, the *a posteriori* error estimation procedure discussed in Section II.E readily extends to unsteady problems by solving the unsteady dual problem, constructing the dual reduced basis $\mathcal{V}_N^{\text{du}}$ using Gappy-POD, and identifying RQ rules using the EQP for the unsteady dual problem and the DWR. We again refer to [37, 38] for details.

C. Parametrized shapes

We now discuss the extension of the RB-RQ method for problems with parametrized shapes (i.e., geometries). We refer to [41] for details and provide an overview of the treatment. To begin, we introduce a diffeomorphism $T(\mu) : \Omega_0 \to \Omega(\mu)$ from the reference domain Ω_0 to the transformed domain $\Omega(\mu)$, where the map is parametrized by $\mu \in \mathcal{D}$. We then introduce an approximation space $\mathcal{V}_{\Omega(\mu)}$ on the transformed (physical) domain and consider the semi-linear form $r_h^{\Omega(\mu)} : \mathcal{V}_{\Omega(\mu)} \times \mathcal{V}_{\Omega(\mu)} \times \mathcal{D} \to \mathbb{R}$ such that

$$r_h^{\Omega(\mu)}(w_h, v_h; \mu) \equiv -\int_{\Omega(\mu)} \nabla v_h : F(w_h; \mu) dx + \int_{\Gamma_I(\mu)} (v_h^+ - v_h^-) \cdot \hat{F}(w_h^+, w_h^-, \hat{n}; \mu) ds + \int_{\Gamma_B(\mu)} v_h^+ \cdot \hat{F}_B(w_h^+, \hat{n}; \mu) ds.$$

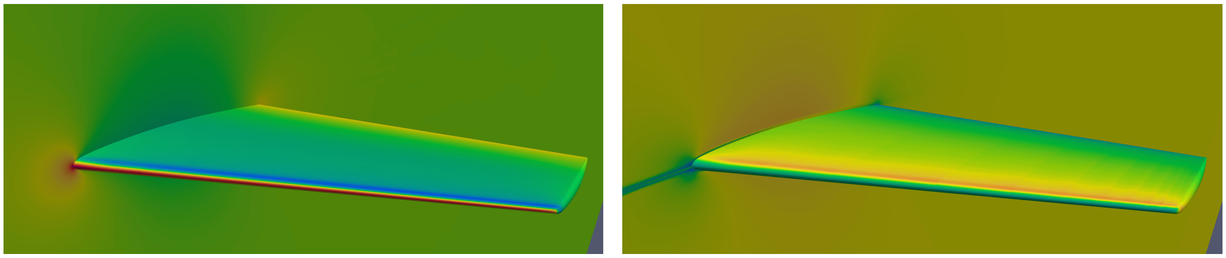
This problem can be recast in the reference domain. Namely, we introduce an approximation space $\mathcal{V}$ on the reference domain Ω_0 and consider the semi-linear form $r_h : \mathcal{V} \times \mathcal{V} \times \mathcal{D} \to \mathbb{R}$ such that

$$\begin{aligned} r_h(w_h, v_h; \mu) \equiv &-\int_{\Omega(\mu)} (J(\mu)^{-T} \nabla v_h) : F(w_h; \mu) |J(\mu)| dx + \int_{\Gamma_I(\mu)} (v_h^+ - v_h^-) \cdot \hat{F}(w_h^+, w_h^-, J(\mu)^{-T} \hat{n}; \mu) |J(\mu)| \|J^{-T} \hat{n}\| ds \\ &+ \int_{\Gamma_B(\mu)} v_h^+ \cdot \hat{F}_B(w_h^+, J(\mu)^{-T} \hat{n}; \mu) |J(\mu)| \|J^{-T} \hat{n}\| ds, \end{aligned}$$

where $J(\mu) : \Omega_0 \to \mathbb{R}^{d\times d}$ is the derivative of the transformation $T(\mu) : \Omega_0 \to \Omega(\mu)$, and $|J(\mu)|$ is its determinant. In words, we recast the problem on parametrized domains as a problem on a (fixed) reference domain with appropriate modifications to the gradients, volumes, and normals. This problem on the (fixed) reference domain fits the form (1), and hence is amenable to approximations by the RB-RQ method. We refer to [41] for details.

D. Problems with moderate- to high-dimensional parameter spaces

We now discuss the extension of the RB-RQ method for high-dimensional problem. High-dimensional problems are challenging because the greedy training set Ξ^{train} and the EQP training set Ξ^{eqp} in Algorithm 1 must be chosen sufficiently large to yield a good coverage of the high-dimensional parameter space $\mathcal{D}$ and to ensure the accuracy of the RB-RQ model, yet small enough so that the offline training is tractable. To address this challenge, we rely on the



(a) primal solution (pressure)

(b) adjoint (continuity equation)

Fig. 3 Example of the primal and adjoint solutions to the ONERA M6 problem.

online-efficient error estimate introduced in Section II.E and in addition introduce adaptive procedures to control the greedy training set Ξ^{train} and the EQP training set Ξ^{eqp}. Our approach is motivated by a high-dimensional sampling approach in [42]. We refer to [41] for details and provide an overview of the extension.

To adaptively construct the greedy training set Ξ^{train}, we update the training set at each greedy iteration so that Ξ^{train} contains parameters for which the RB-RQ model yields largest errors (i.e., the model works poorly). Specifically, we retain $K < L \equiv |\Xi^{\text{train}}|$ with largest errors from the previous greedy iterations and augment the training set with $L - K$ new random points, where $L \equiv |\Xi^{\text{train}}|$ is the cardinality of Ξ^{train}. In other words, the training set in the i-th greedy iteration, $\Xi^{\text{train},i}$, is

$$\Xi^{\text{train},i} = \Xi_K^{\text{train},i-1} \cup \Xi_{L-K}^{\text{rand}},$$

where $\Xi_K^{\text{train},i-1}$ is the K points with the largest error estimate $\tilde{\eta}_{N-1}^{\text{rb}}$, and Ξ_{L-K}^{rand} is a set of $L - K$ points randomly sampled from $\mathcal{D}$. The procedure yields a sequence of greedy training sets $\Xi^{\text{train},i}$, $i = 1, 2, 3, \ldots$, that focus on high-error regions, which must be sampled to reduce the RB-RQ error.

To adaptively construct the EQP training set Ξ^{eqp}, we consider an hierarchical enrichment of the set. Namely, in each greedy iteration, we first initialize $\Xi^{\text{eqp}} \equiv \Xi^{\text{rb}}$ so that $|\Xi^{\text{eqp}}| = N$ regardless of the size of the greedy training set. We next solve the quadrature optimization problem (9) for the current set Ξ^{eqp} and an enriched set $\Xi^{\text{eqp}\prime} \supset \Xi^{\text{eqp}}$ to construct two RB-RQ models associated with the two training sets. We then evaluate the RB-RQ output for all training points in Ξ^{train} using the two models. We compare the difference in the RB-RQ outputs evaluated by the two models to check if the difference is less than the user-prescribed EQP tolerance. If not, we repeat the procedure using the (new) current set $\Xi^{\text{eqp}\prime}$ and a new enriched set $\Xi^{\text{eqp}\prime\prime} \supset \Xi^{\text{eqp}\prime}$, which is enriched with the parameter points that yield large RQ errors. We refer to [41] for details.

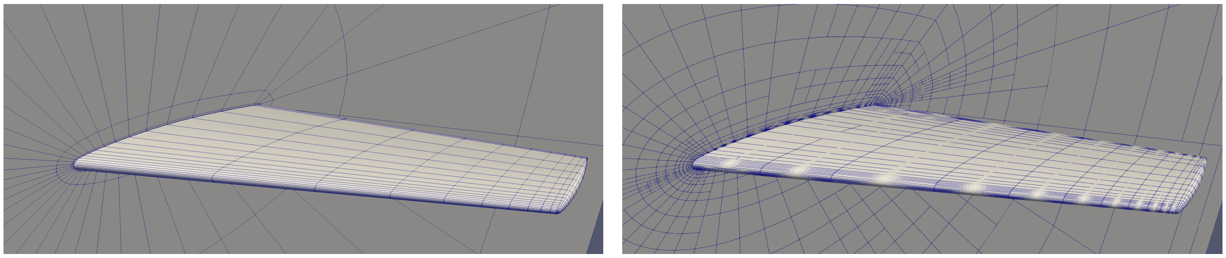
IV. Examples

We now demonstrate the construction and evaluation of RB-RQ models using several aerodynamics problems.

A. Reynolds-averaged Navier-Stokes flow over ONERA M6

We first consider turbulent flow over the ONERA M6 wing. The governing equation is the Reynolds-averaged Navier-Stokes (RANS) equations with the Spalart-Allmaras (SA) turbulence model [43] and the SA-neg correction [44]. The two parameters are the freestream Mach number $M_\infty \in [0.3, 0.4]$ and the angle of attack $\alpha \in [0°, 2°]$. The Reynolds number is fixed at $Re_c = 10^6$. The quantity of interest is the drag on the wing. Figure 3 shows the primal and adjoint solutions for the centroid parameter values of $M_\infty = 0.35$ and $\alpha = 1°$. We wish to construct a RB-RQ model with both the FE and RB-RQ error of less than 0.5%, so that the combined error is less than 1%.

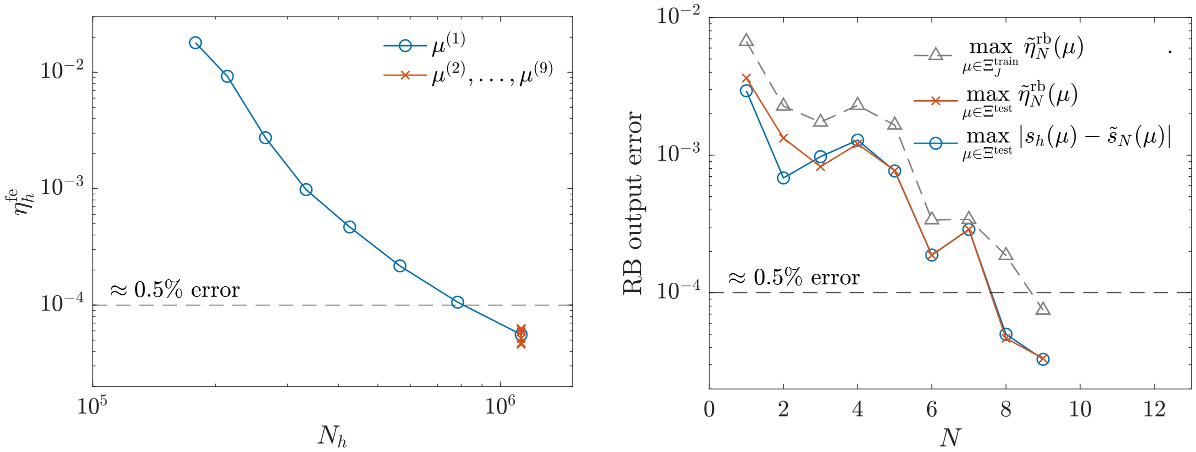
We now invoke Algorithm 1 to construct a RB-RQ model that meets the user-specified error tolerance. The initial $\mathbb{P}^2$ DG discretization has 178560 degrees of freedom, counting degrees of freedom associated with the continuity, momentum, energy, and SA equations separately. An output-based adaptive DG solver is used to reduce the FE output error to less than 0.5%; the resulting DG discretization has $N_h = 1122420$ degrees of freedom and $Q_h = 6307992$ quadrature points. Figure 4 shows the initial and final meshes. Figure 5(a) summarizes the FE error convergence behavior; all of the mesh adaptation is performed for the first parameter value, and the subsequent parameter values do not require additional refinements.



(a) initial mesh

(b) adapted mesh

Fig. 4 Initial mesh and adapted mesh for the ONERA M6 problem.



(a) FE error

(b) RB error

Fig. 5 Convergence of the FE and RB-RQ errors for the ONERA M6 problem.

Figure 5(b) shows the RB-RQ error convergence behavior during offline training. The RB-RQ discretization is trained using a (static) training set of the size $|\Xi^{\mathrm{train}}| = 5 \times 5$, which is equidistributed over the Mach number and angle of attack range. In the greedy training, the maximum DWR error estimate over the training set Ξ^{train}, $\max_{\mu \in \Xi^{\mathrm{train}}} |\tilde{\eta}_N^{\mathrm{rb}}(\mu)|$, converges to within 0.5% error for $N = 9$ snapshots (i.e., using the information from nine adaptive DG solves). We also test the RB-RQ discretization against a test set Ξ^{test} of five random parameter values not in the training set Ξ^{train}. Figure 5(b) shows that the maximum output error over the test set, $\max_{\mu \in \Xi^{\mathrm{test}}} |s_h(\mu) - \tilde{s}_N(\mu)|$, is less than 0.5% for $N \geq 8$. We also observe that the maximum DWR error estimate for the test set, $\max_{\mu \in \Xi^{\mathrm{test}}} |\tilde{\eta}_N^{\mathrm{rb}}(\mu)|$, closely follows the true error and hence the error estimate is effective. Table 1 summarizes the properties of the RB-RQ discretization. Relative to the DG discretization, the RB-RQ discretization reduces the number of degrees of freedom from $N_h = 1122420$ to $N = 9$ and the number of residual, output, and DWR quadrature points from $Q_h = 6307992$ to $\tilde{Q}_{\mathrm{res}} = 153$, $\tilde{Q}_{\mathrm{out}} = 12$, and $\tilde{Q}_{\mathrm{dwr}} = 177$, respectively.

	degrees of freedom	quadrature points			wall-clock speedup	
	N	Q_{res}	Q_{out}	Q_{dwr}	soln. only	w/ err. est.
DG-FEM	1122420	6307992	139680	6307992	-	-
RB-RQ	9	153	12	177	22377	21267

Table 1 Summary of computational reduction for the ONERA M6 problem. The RB-RQ discretization achieves less than 0.5% output error for all training and test parameter values.

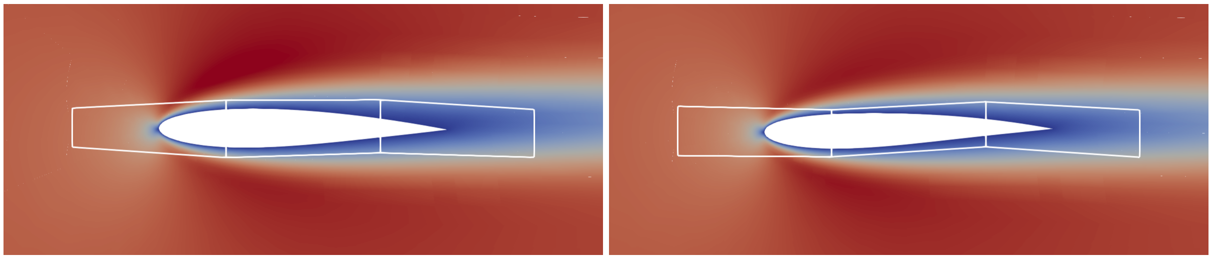


Fig. 6 Examples of solutions (Mach number) to the shape-parametrized airfoil problem for two different geometries.

We finally report the computational time. We use the wall-clock time required to solve the DG-FEM problem on the final adapted mesh on a 80-core cluster as the time unit and denote it by t_{fe}. The wall-clock computational time for the entire offline stage is $\approx 59 t_{\text{fe}}$. The majority (53%) of the offline computational time is spent on the initial adaptive mesh refinement and the computation of the solution and the associated error estimates for the nine snapshots. The next dominant cost (46%) is the computation of the RQ rules using EQP, which requires multiple evaluation of the DG residual for training parameter values in Ξ^{train}. In the online stage, the RB-RQ model provides a significant computational speedup; the online computational time is $1/22377 \times t_{\text{fe}}$ for the output only and $1/21267 \times t_{\text{fe}}$ for both the output and the DWR *a posteriori* error estimate. We note that the computational speedup is relative to the adaptive high-order DG method, which is significantly more efficient than the lower-order and/or non-adaptive method. In addition, this wall-clock speedup is conservative, as the small RB-RQ model is incapable of fully leveraging the 80-core cluster; the speedup in CPU time would be more significant than the wall-clock speedup reported.

B. Laminar flow over shape-parametrized airfoils

We next consider laminar flow over a family of shape-parametrized airfoils based on NACA0012. The governing equation is the (laminar) Navier-Stokes equations in entropy variables [7]. We fix the Mach number and Reynolds number to $Re_c = 5000$ and $M_\infty = 0.4$, respectively. The airfoil is transformed using cubic spline interpolation with 4×2 control points over the $1.6c \times 0.16c$ rectangular region centered about the airfoil, as shown in Figure 6. We allow each of the eight control points to vary in the x_2 direction by a distance of $\Delta x_2 \in [-0.02c, 0.02c]$, where c is the chord. We in addition parametrize the angle of attack so that $\alpha \in [1^\circ, 2^\circ]$. We hence have eight geometry degrees of freedom and one angle of attack degree of freedom, which yields the parameter space $\mathcal{D} \equiv [-0.02, 0.02]^8 \times [1^\circ, 2^\circ] \subset \mathbb{R}^9$. The quantity of interest is the drag on the airfoil. Examples of shape transformations are shown in Figure 6.

We now construct a RB-RQ discretization using Algorithm 1, with the extensions to geometry-parametrized and high-dimensional problems summarized in Sections III.C and III.D, respectively. The initial $\mathbb{P}^2$ DG discretization has 9600 degrees of freedom, counting degrees of freedom associated with the continuity, momentum, and energy equations separately. An output-based adaptive DG solver is used to reduce the FE output error to less than 0.1%; the resulting DG discretization has $N_h = 15840$ degrees of freedom and $Q_h = 32322$ quadrature points.

Figure 7 shows the RB-RQ error convergence behavior during offline training. The RB-RQ discretization is trained using an adaptively enriched greedy training set of the size $|\Xi^{\text{train}}| = 2000$. We use a much larger training set than the ONERA M6 case considered in Section IV.A to treat the higher-dimensional parameter space. In the greedy training, the maximum DWR error estimate over the training set Ξ^{train}, $\max_{\mu \in \Xi^{\text{train}}} |\tilde{\eta}_N^{\text{rb}}(\mu)|$, converges to within 1% error for $N = 16$ snapshots (i.e., using the information from 16 DG solves). The average size of the adaptively selected EQP training set is $|\Xi^{\text{EQP}}| \approx 29$, which is significantly smaller than the greedy training set of the size $|\Xi^{\text{train}}| = 2000$. The smaller EQP training set enables efficient construction of the RQ rules.

We also test the RB-RQ discretization against a test set Ξ^{test} of 10 random parameter values not in the training set Ξ^{train}. Figure 7 shows that the maximum output error over the test set, $\max_{\mu \in \Xi^{\text{test}}} |s_h(\mu) - \tilde{s}_N(\mu)|$, is less than 1% for $N \geq 13$. We also observe that the maximum DWR error estimate for the test set, $\max_{\mu \in \Xi^{\text{test}}} |\tilde{\eta}_N^{\text{rb}}(\mu)|$, closely follows the true error and hence the error estimate is effective. The online-efficient *a posteriori* error estimate enables the use of the large adaptively enriched greedy training set of the size $|\Xi^{\text{train}}| = 2000$, which provides a sufficient coverage of the nine-dimensional parameter space.

Table 2 summarizes the properties of the RB-RQ discretization. Relative to the DG discretization, the RB-RQ discretization reduces the number of degrees of freedom from $N_h = 15840$ to $N = 16$ and the number of residual,

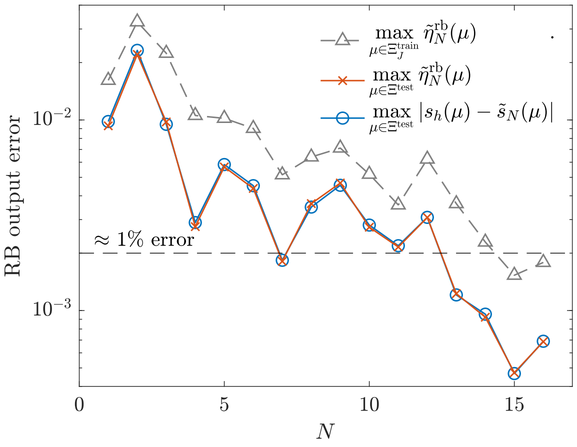


Fig. 7 Convergence of the RB-RQ error for the shape-parametrized airfoil problem.

	degrees of freedom	quadrature points			wall-clock speedup	
	N	Q_{res}	Q_{out}	Q_{dwr}	soln. only	w/ err. est.
DG-FEM	15840	32322	606	32322	-	-
RB-RQ	16	423	24	456	416	386

Table 2 Summary of computational reduction for the shape-parametrized airfoil problem. The RB-RQ discretization achieves less than 1% output error for all training and test parameter values.

output, and DWR quadrature points from $Q_h = 32322$ to $\tilde{Q}_{\text{res}} = 423$, $\tilde{Q}_{\text{out}} = 24$, and $\tilde{Q}_{\text{dwr}} = 456$, respectively.

We finally report the computational time. We use the wall-clock time required to solve the DG-FEM problem on the final adapted mesh on a 6-core workstation as the time unit and denote it by t_{fe}. The wall-clock computational time for the entire offline stage is $167 t_{\text{fe}}$. The majority of the offline computational time is spent on the construction of the RQ rule using the adaptive EQP algorithm for high-dimensional parameter spaces; we are currently working on improving the efficiency of the algorithm to reduce the offline computational cost. In the online stage, the RB-RQ model provides a significant computational speedup; the online computational time is $1/416 \times t_{\text{fe}}$ for the output only and $1/386 \times t_{\text{fe}}$ for both the output and the DWR *a posteriori* error estimate.

C. Uncertainty quantification (UQ) of RANS flow with the Spalart-Allmaras turbulence model

We next consider turbulent flow over the RAE2822 airfoil. The governing equation is the RANS equations with the Spalart-Allmaras (SA) turbulence model [43] and the SA-neg correction [44]. The Mach number, the Reynolds number, and the angle of attack are fixed at $M_\infty = 0.3$, $Re_c = 6.5 \times 10^6$, and $\alpha = 2°$, respectively. We wish to quantify the effect of the parametric uncertainty in the SA model. Based on the UQ study of [45], we consider the following seven uncertain parameters and the associated intervals for the fully turbulent RANS-SA equations: $\sigma \in [0.6, 1.0]$, $\kappa \in [0.38, 0.42]$, $c_{v_2} \in [6.9, 7.3]$, $c_{w_2} \in [0.055, 0.3525]$, $c_{w_3} \in [1.75, 2.5]$, $c_{b_1} \in [0.12893, 0.137]$, and $c_{b_2} \in [0.60983, 0.6875]$. We assume the seven parameters are independent and uniformly distributed. We wish to quantify the uncertainty in the drag due to the variabilities in these input parameters. Figure 8 shows representative flow solutions; the figure illustrates the UQ process using the Monte-Carlo method.

Before we apply the RB-RQ discretization to the problem, we make a few cautionary remarks. First, this problem can be solved using, for example, sparse grid and (non-reduced) CFD analysis, as done in [45, 46]. Second, [45] notes that there exist implicit relationships between SA model parameters, which make c_{b_1}, c_{b_2}, c_{w_2} and c_{v_1} functions of σ and κ, and reduce the parametric dimension of the problem from seven to three. As our goal in this example is to illustrate how model reduction enables reliable UQ of problems with a moderate number of input parameters, we treat all seven parameters as independent variables.

We now construct a RB-RQ discretization using Algorithm 1, with the extension for high-dimensional problems summarized in Section III.D. The initial $\mathbb{P}^2$ DG discretization has 15180 degrees of freedom, counting degrees of

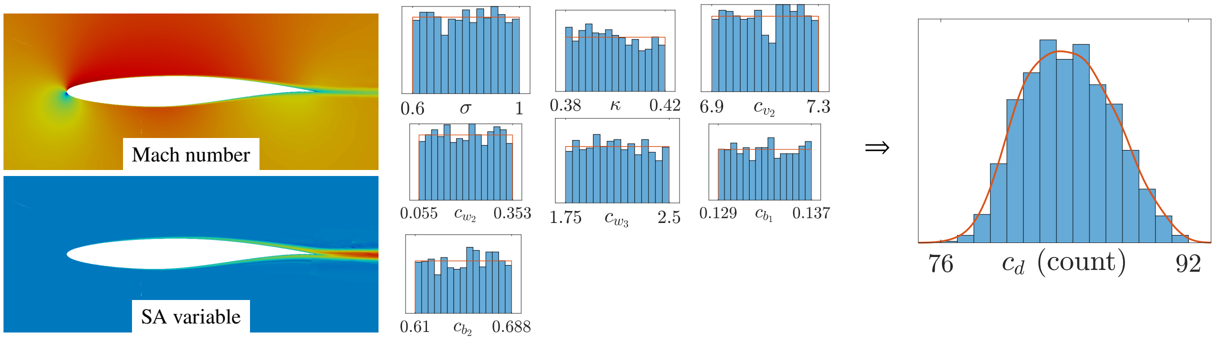
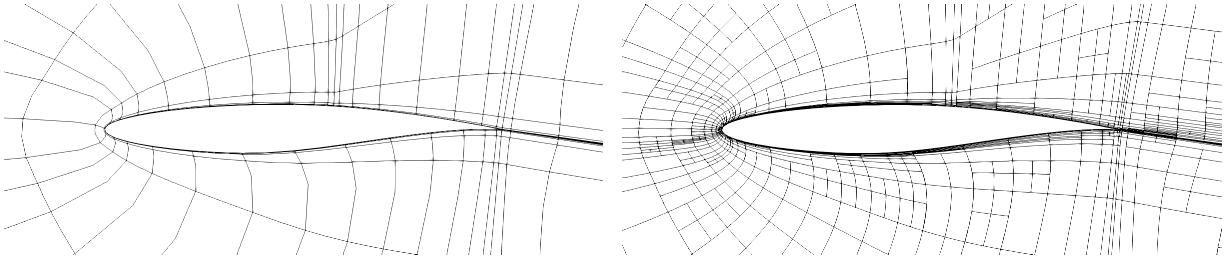


Fig. 8 Illustration of uncertainty quantification of RANS-SA flow over RAE2822.



(a) initial mesh

(b) adapted mesh

Fig. 9 Initial mesh and adapted mesh for uncertainty quantification of RANS-SA flow over RAE2822.

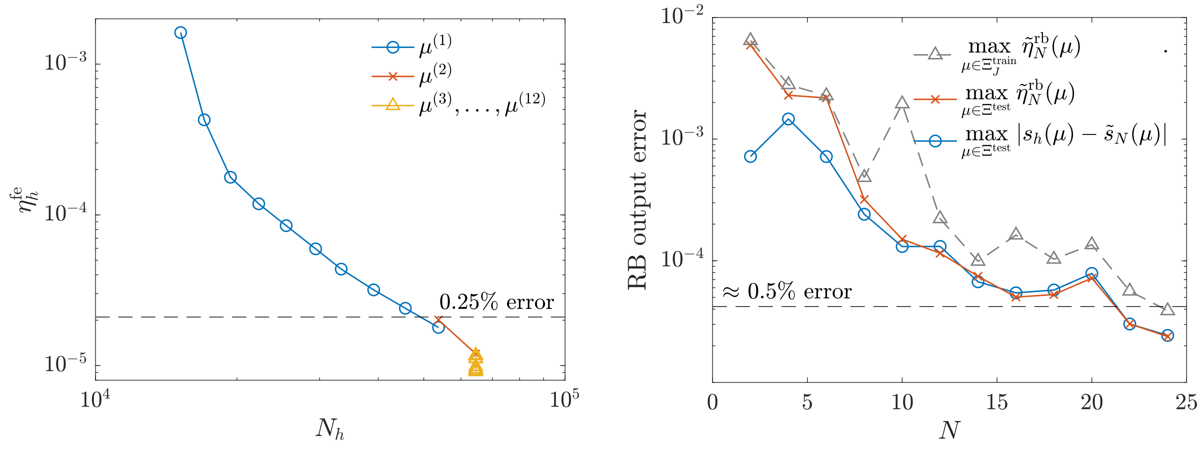
freedom associated with the continuity, momentum, energy, and SA equations separately. An output-based adaptive DG solver is used to reduce the FE output error to less than 0.25%; the resulting DG discretization has $N_h = 64530$ degrees of freedom and $Q_h = 104274$ quadrature points. Figure 9 show the initial and final meshes. Figure 10(a) summarizes the FE convergence behavior; the majority of the mesh refinement is performed for the first parameter value, but one additional mesh refinement is performed for the second parameter value to meet the user-prescribed FE drag error tolerance of 0.25%.

Figure 10(b) shows the RB-RQ error convergence behavior during offline training. The RB-RQ discretization is trained using a greedy training set of the size $|\Xi^{\text{train}}| = 400$, which are randomly sampled from the seven-dimensional parameter space $\mathcal{D}$. In the greedy training, the maximum DWR error estimate over the training set Ξ^{train}, $\max_{\mu\in\Xi^{\text{train}}} |\tilde{\eta}_N^{\text{rb}}(\mu)|$, converges to within 0.5% drag error using $N = 12$ snapshots (i.e., using the information from 12 DG solves). We note that the size of the RB is $N = 24$ because we split each snapshot into components associated with the mean-flow equations and the SA equation, and hence each snapshot yields two basis functions.

We also test the RB-RQ discretization against a test set Ξ^{test} of 10 random parameters not in the training set Ξ^{train}. Figure 10(b) shows the maximum output error over the test set, $\max_{\mu\in\Xi^{\text{test}}} |s_h(\mu) - \tilde{s}_N(\mu)|$, is less than 0.5% for $N > 22$. We also observe that the maximum DWR error estimate for the test set, $\max_{\mu\in\Xi^{\text{test}}} |\tilde{\eta}_N^{\text{rb}}(\mu)|$, closely follows the true error and hence the error estimate is effective.

Table 3 summarizes the properties of the RB-RQ model. Relative to the DG discretization, the RB-RQ discretization reduces the number of degrees of freedom from $N_h = 64530$ to $N = 24$ and the number of residual, output, and DWR quadrature points from $Q_h = 105432$ to $\tilde{Q}_{\text{res}} = 522$, $\tilde{Q}_{\text{out}} = 6$, and $\tilde{Q}_{\text{dwr}} = 214$, respectively.

We finally report the computational time. We use the wall-clock time required to solve the DG-FEM problem on the final adapted mesh on a 6-core workstation as the time unit and denote it by t_{fe}. The wall-clock computational time for the entire offline stage is $215 t_{\text{fe}}$. Similarly to the shape-parametrized airfoil case in Section IV.B, the majority of the offline computational time is spent on the construction of the RQ rules using the adaptive EQP algorithm for the high-dimensional parameter space; we are currently working on improving the efficiency of the algorithm to reduce the offline computational cost. In the online stage, the RB-RQ model provides a significant computational speedup; the online computational time is $1/303 \times t_{\text{fe}}$ for the output only and $1/294 \times t_{\text{fe}}$ for both the output and the DWR *a posteriori* error estimate.



(a) FE error

(b) RB-RQ error

Fig. 10 Convergence of the FE and RB-RQ errors for uncertainty quantification of RANS-SA flow over RAE2822.

	degrees of freedom	quadrature points			speedup	
	N	Q_{res}	Q_{out}	Q_{dwr}	soln. only	w/ err. est.
DG-FEM	64530	105432	756	105432	-	-
RB-RQ	24	522	6	214	303	294

Table 3 Summary of computational reduction for the uncertainty quantification of RANS-SA flow over RAE2822. The RB-RQ discretization achieves less than 0.5% drag count error for all training and test parameter values.

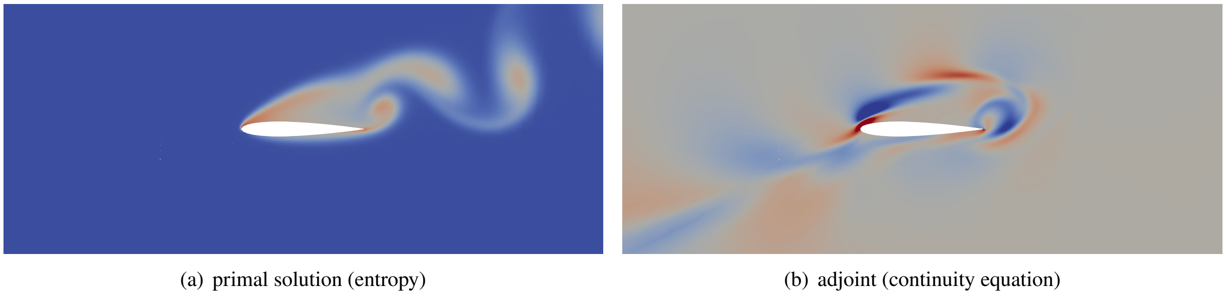
In the context of UQ, the speed up factor of 300 implies that we can perform 300 "brute force" Monte Carlo solves using the RB-RQ model in the time it takes to perform a single DG solve, once the RB-RQ discretization is constructed. In addition, the RB-RQ model meets the tight error tolerance and provides an *a posteriori* error estimate in the predictive setting, which ensures that the use of the reduced model does not negatively impact the accuracy of the UQ process. We refer to [46, 47] for discussions of the importance of controlling the discretization error for a reliable UQ of aerodynamics problems, which exhibit a wide range of scales.

D. Unsteady flow over NACA0012

We now consider unsteady laminar flow past NACA0012. The governing equation is the laminar Navier-Stokes equation in entropy variables [7]. The angle of attack is fixed at $\alpha = 20°$ and the Reynolds number is $Re_c \in [300, 600]$. The initial condition for the case is a snapshot of a (nearly) periodic solution for $Re_c = 450$. The quantity of interest is the time-averaged drag over a window of 20 nondimesionalized time units, which corresponds to approximate 2 periods of the (nearly) periodic solution. Figure 11 shows snapshots of the primal and adjoint solutions.

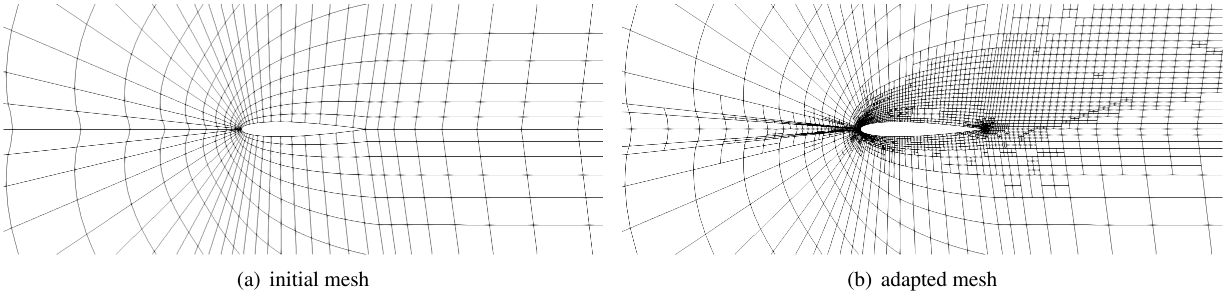
We now invoke Algorithm 1, with the extension to unsteady problems summarized in Section III.B. The initial $\mathbb{P}^2$ DG discretization has 31104 degrees of freedom, counting degrees of freedom associated with the continuity, momentum, and energy equations separately. The time integration is performed using a three-stage DIRK method [40]. An output-based adaptive DG solver is used to solve for the initial transient and then time-window of interest such that the error in the FE output is less than 0.05%; the resulting DG discretization has $N_h = 155904$ degrees of freedom and $Q_h = 318036$ quadrature points. Figure 12 show the initial and final meshes. Figure 13(a) summarizes the FE convergence behavior; adaptive mesh refinement is performed for the initial transient, the first parameter value, and the second parameter value.

Figure 13(b) shows the RB-RQ error convergence behavior during the offline training. The RB-RQ discretization is trained using a greedy training set of the size $|\Xi^{\mathrm{train}}| = 30$, which are equidistributed over the Reynolds number range. In


(a) primal solution (entropy)

(b) adjoint (continuity equation)

Fig. 11 Snapshots of the primal and adjoint solutions to the unsteady flow past NACA0012.


(a) initial mesh

(b) adapted mesh

Fig. 12 Initial mesh and adapted mesh for the unsteady flow past NACA0012.

the greedy training, the greedy algorithm evaluates the unsteady DG solution for three different parameter values. The POD-Greedy updates of these solutions yield a reduced basis of the size $N = 42$. The maximum DWR error estimate over the training set Ξ^{train}, $\max_{\mu\in\Xi^{\text{train}}} |\tilde{\eta}_N^{\text{rb}}(\mu)|$, converges to less than 0.5‰.

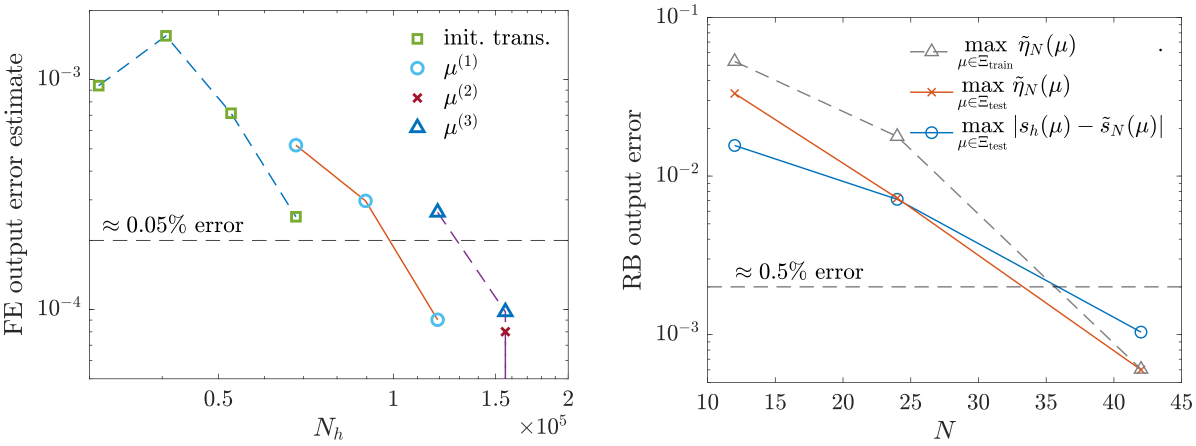
We also test the RB-RQ discretization against a test set Ξ^{test} of five random parameters not in the training set Ξ^{train}. Figure 13(b) shows that the maximum output error over the test set, $\max_{\mu\in\Xi^{\text{test}}} |s_h(\mu) - \tilde{s}_N(\mu)|$, is less than 0.5‰. We also observe that the maximum DWR error estimate for the test set, $\max_{\mu\in\Xi^{\text{test}}} |\tilde{\eta}_N^{\text{rb}}(\mu)|$, closely follows the true error and hence the error estimate is effective for this unsteady problem (though not as effective as for the steady problems considered in this work).

Table 4 summarizes the performance of the RB-RQ model. Relative to the DG discretization, the RB-RQ discretization reduces the number of primal and dual degrees of freedom from $N_h = 155904$ to $N = 42$ and $N_{\text{du}} = 24$, respectively, and the number of residual, output, and DWR quadrature points from $Q_h = 318036$ to $\tilde{Q}_{\text{res}} = 1267$, $\tilde{Q}_{\text{out}} = 35$, and $\tilde{Q}_{\text{dwr}} = 799$, respectively.

We finally report the computational time. We use the wall-clock time required to solve the FE problem on the final adapted mesh on a 40-core cluster as the time unit and denote it by t_{fe}. The wall-clock computational time for the entire offline stage is $69t_{\text{fe}}$. In the online stage, the RB-RQ model provides a significant computational speedup; the online computational time is $1/328 \times t_{\text{fe}}$ for the output only and $1/248 \times t_{\text{fe}}$ for both the output and the DWR *a posteriori* error estimate. The computation of the online-efficient *a posteriori* error estimate is relatively more expensive for the unsteady problem than for steady problems, as the former requires the solution of the unsteady dual solution.

	degrees of freedom		quadrature points			speedup	
	N	N_{du}	Q_{res}	Q_{out}	Q_{dwr}	soln. only	w/ err. est.
DG-FEM	155904	155904	318036	2202	318036	-	-
RB-RQ	42	24	1267	35	799	328	248

Table 4 Summary of computational reduction for the unsteady flow past NACA0012. The RB-RQ discretization achieves less than 0.5‰ output error for all training and test parameter values.



(a) FE error

(b) RB-RQ error

Fig. 13 Convergence of the FE and RB-RQ errors for the unsteady flow past NACA0012.

V. Summary and conclusions

This work presents an approach to reliable and efficient projection-based model reduction for many-query/real-time parametrized (nonlinear) aerodynamics problems. In the offline stage, a RB-RQ model that meets the user-prescribed output error tolerance is constructed using the simultaneous FE, RB, and RQ training algorithm (Algorithm 1). The algorithm controls the FE error using adaptive high-order DG method, the RB error through the successive enrichment of the RB space using the greedy algorithm, and the RQ error using the EQP. Through this adaptive control of the various discretization errors, the accurate RB-RQ model is constructed without any user interventions. In the online stage, the RB-RQ model provides rapid solution for many different parameter values. The RB-RQ model in addition equips each output prediction with an online-efficient DWR *a posteriori* error estimate, so that the model can be used confidently even for parameter values for which the model was not trained (i.e., in the "predictive" setting). We demonstrated the efficacy and versatility of the framework using four aerodynamics problems: RANS flow over the ONERA M6 wing with the Mach number and the angle of attack as the parameters; laminar flow over shape-parametrized airfoils; UQ of RANS flow with variabilities in the empirical parameters of the SA turbulence mode; and unsteady separated flow past NACA0012 with the Reynolds number as the parameter. We observed typical wall-clock speedup of ∼300–20000 and accurate *a posteriori* error estimate in predictive settings.

While the examples have shown some promising results, there remains a number of challenges to model reduction of complex aerodynamics flows. The first challenge is the treatment of shock waves and more generally parameter-dependent irregular features. It is provably impossible to construct a low-dimensional linear (RB) space that can approximate solutions containing parameter-dependent irregular features [6]. One way to overcome the challenge is to consider nonlinear approximation spaces, as considered for simple problems in for example [48, 49], but further progress is necessary to use the technique in complex aerodynamics problems. The second challenge is the further reduction of the offline training cost. In this work the offline training cost was reduced by employing adaptive algorithms, including adaptive mesh refinement to compute the snapshots and the greedy algorithm to judiciously chose the training parameter values, but the training cost is still 50 to 200 times the cost of a single DG-FEM flow solve on the final adapted mesh. A further reduction in the offline training cost is required to apply the method to industrial-scale problems with a large number of parameters in practical engineering settings.

Acknowledgments

The financial support for this work was provided by the Natural Sciences and Engineering Research Council of Canada and by the Ontario Graduate Scholarship program. Some of the computations were performed on the Niagara supercomputer at the SciNet HPC Consortium. SciNet is funded by the Canada Foundation for Innovation; the Government of Ontario; Ontario Research Fund - Research Excellence; and the University of Toronto.

References

[1] Rozza, G., Huynh, D. B. P., and Patera, A. T., “Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations — Application to transport and continuum mechanics,” *Archives of Computational Methods in Engineering*, Vol. 15, No. 3, 2008, pp. 229–275. https://doi.org/10.1007/s11831-008-9019-9.

[2] Benner, P., Gugercin, S., and Willcox, K., “A survey of projection-based model reduction methods for parametric dynamical systems,” *SIAM Review*, Vol. 57, No. 4, 2015, pp. 483–531. https://doi.org/10.1137/130932715.

[3] Hesthaven, J. S., Rozza, G., and Stamm, B., *Certified reduced basis methods for parametrized partial differential equations*, Springer, 2016. https://doi.org/10.1007/978-3-319-22470-1.

[4] Lucia, D. J., Beran, P. S., and Silva, W. A., “Reduced-order modeling: new approaches for computational physics,” *Progress in Aerospace Sciences*, Vol. 40, No. 1-2, 2004, pp. 51–117. https://doi.org/10.1016/j.paerosci.2003.12.001.

[5] de C. Henshaw, M. J., Badcock, K. J., Vio, G. A., Allen, C. B., Chamberlain, J., Kaynes, I., Dimitriadis, G., Cooper, J. E., Woodgate, M. A., Rampurawala, A. M., Jones, D., Fenwick, C., Gaitonde, A. L., Taylor, N. V., Amor, D. S., Eccles, T. A., and Denley, C. J., “Non-linear aeroelastic prediction for aircraft applications,” *Progress in Aerospace Sciences*, Vol. 43, No. 4-6, 2007, pp. 65–137. https://doi.org/10.1016/j.paerosci.2007.05.002.

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

[7] Barth, T. J., “Numerical methods for gasdynamic systems on unstructured meshes,” *An Introduction to Recent Developments in Theory and Numerics for Conservation Laws*, edited by D. Kröner, M. Ohlberger, and C. Rohde, Springer-Verlag, 1999, pp. 195–282. https://doi.org/10.1007/978-3-642-58535-7_5.

[8] Cockburn, B., “Discontinuous Galerkin methods,” *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik*, Vol. 83, No. 11, 2003, pp. 731–754. https://doi.org/10.1002/zamm.200310088.

[9] Hesthaven, J. S., and Warburton, T., *Nodal discontinuous Galerkin methods*, Springer New York, 2008. https://doi.org/10.1007/978-0-387-72067-8.

[10] Becker, R., and Rannacher, R., “An optimal control approach to a posteriori error estimation in finite element methods,” *Acta Numerica*, Vol. 10, 2001, pp. 1–102. https://doi.org/10.1017/s0962492901000010.

[11] Ainsworth, M., and Oden, J. T., “A posteriori error estimation in finite element analysis,” *Computer Methods in Applied Mechanics and Engineering*, Vol. 142, 1997, pp. 1–88. https://doi.org/10.1002/9781118032824.

[12] Fidkowski, K., and Darmofal, D., “Review of output-based error estimation and mesh adaptation in computational fluid dynamics,” *AIAA Journal*, Vol. 49, No. 4, 2011, pp. 673–694. https://doi.org/10.2514/1.j050073.

[13] Veroy, K., Prud’homme, C., Rovas, D., and Patera, A., “A posteriori error bounds for reduced-basis approximation of parametrized noncoercive and nonlinear elliptic partial differential equations,” *16th AIAA Computational Fluid Dynamics Conference*, American Institute of Aeronautics and Astronautics, 2003. https://doi.org/10.2514/6.2003-3847.

[14] An, S. S., Kim, T., and James, D. L., “Optimizing cubature for efficient integration of subspace deformations,” *ACM Trans. Graph.*, Vol. 27, No. 5, 2008, pp. 165:1–165:10. https://doi.org/10.1145/1409060.1409118, URL http://doi.acm.org/10.1145/1409060.1409118.

[15] Farhat, C., Avery, P., Chapman, T., and Cortial, J., “Dimensional reduction of nonlinear finite element dynamic models with finite rotations and energy-based mesh sampling and weighting for computational efficiency,” *International Journal for Numerical Methods in Engineering*, Vol. 98, No. 9, 2014, pp. 625–662. https://doi.org/10.1002/nme.4668.

[16] Chapman, T., Avery, P., Collins, P., and Farhat, C., “Accelerated mesh sampling for the hyper reduction of nonlinear computational models,” *International Journal for Numerical Methods in Engineering*, Vol. 109, No. 12, 2017, pp. 1623–1654. https://doi.org/10.1002/nme.5332.

[17] Patera, A. T., and Yano, M., “An LP empirical quadrature procedure for parametrized functions,” *Comptes Rendus Mathematique*, Vol. 355, No. 11, 2017, pp. 1161–1167. https://doi.org/10.1016/j.crma.2017.10.020.

[18] Yano, M., and Patera, A. T., “An LP empirical quadrature procedure for reduced basis treatment of parametrized nonlinear PDEs,” *Computer Methods in Applied Mechanics and Engineering*, Vol. 344, 2019, pp. 1104–1123. https://doi.org/10.1016/j.cma.2018.02.028.

[19] Yano, M., "Discontinuous Galerkin reduced basis empirical quadrature procedure for model reduction of parametrized nonlinear conservation laws," *Advances in Computational Mathematics*, Vol. 45, No. 5-6, 2019, pp. 2287–2320. https://doi.org/10.1007/s10444-019-09710-z.

[20] Yano, M., "Goal-oriented model reduction of parametrized nonlinear PDEs; application to aerodynamics," *International Journal for Numerical Methods in Engineering*, Vol. accepted, 2020.

[21] LeGresley, P., and Alonso, J., "Airfoil design optimization using reduced order models based on proper orthogonal decomposition," *Fluids 2000 Conference and Exhibit*, American Institute of Aeronautics and Astronautics, 2000. https://doi.org/10.2514/6.2000-2545.

[22] LeGresley, P. A., and Alonso, J. J., "Investigation of non-linear projection for POD based reduced order models for aerodynamics," *39th Aerospace Sciences Meeting and Exhibit*, American Institute of Aeronautics and Astronautics, 2001. https://doi.org/10.2514/6.2001-926.

[23] LeGresley, P. A., and Alonso, J. J., "Dynamic domain decomposition and error correction for reduced order models," *41st Aerospace Sciences Meeting and Exhibit*, American Institute of Aeronautics and Astronautics, 2003. https://doi.org/10.2514/6.2003-250.

[24] Washabaugh, K., Zahr, M. J., and Farhat, C., "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. https://doi.org/10.2514/6.2016-1814.

[25] Zimmermann, R., and Görtz, S., "Non-linear reduced order models for steady aerodynamics," *Procedia Computer Science*, Vol. 1, No. 1, 2012, pp. 165–174. https://doi.org/https://doi.org/10.1016/j.procs.2010.04.019, iCCS 2010.

[26] Zimmermann, R., and Görtz, S., "Improved extrapolation of steady turbulent aerodynamics using a non-linear POD-based reduced order model," *The Aeronautical Journal*, Vol. 116, No. 1184, 2012, pp. 1079–1100. https://doi.org/10.1017/s0001924000007491.

[27] Tezaur, I., Fike, J., Carlberg, K., Barone, M., Maddix, D., Mussoni, E., and Balajewicz, M., "Advanced fluid reduced order models for compressible flow," Tech. rep., Sandia National Laboratories, 2017.

[28] Yano, M., "Model reduction in computational aerodynamics," *Model Order Reduction*, De Gruyter, 2020, pp. 201–236. https://doi.org/10.1515/9783110499001-006.

[29] Hartmann, R., and Houston, P., "Error estimation and adaptive mesh refinement for aerodynamic flows," *VKI LS 2010-01: 36th CFD/ADIGMA course on hp-adaptive and hp-multigrid methods, Oct. 26-30, 2009*, edited by H. Deconinck, Von Karman Institute for Fluid Dynamics, Rhode Saint Genèse, Belgium, 2009.

[30] Roe, P. L., "Approximate Riemann solvers, parameter vectors, and difference schemes," *Journal of Computational Physics*, Vol. 43, No. 2, 1981, pp. 357–372. https://doi.org/10.1006/jcph.1997.5705.

[31] Kelley, C. T., and Keyes, D. E., "Convergence analysis of pseudo-transient continuation," *SIAM Journal on Numerical Analysis*, Vol. 35, No. 2, 1998, pp. 508–523. https://doi.org/10.1137/s0036142996304796.

[32] Saad, Y., and Schultz, M. H., "GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems," *SIAM Journal on scientific and statistical computing*, Vol. 7, No. 3, 1986, pp. 856–869. https://doi.org/10.1137/0907058.

[33] Persson, P.-O., and Peraire, J., "Newton-GMRES preconditioning for discontinuous Galerkin discretizations of the Navier-Stokes equations," *SIAM Journal on Scientific Computing*, Vol. 30, No. 6, 2008, pp. 2709–2733. https://doi.org/10.1137/070692108.

[34] Yano, M., Modisette, J. M., and Darmofal, D., "The importance of mesh adaptation for higher-order discretizations of aerodynamic flows," AIAA 2011-3852, AIAA, 2011. https://doi.org/10.2514/6.2011-3852.

[35] Arnold, D. N., Brezzi, F., Cockburn, B., and Marini, L. D., "Unified analysis of discontinuous Galerkin methods for elliptical problems," *SIAM Journal on Numerical Analysis*, Vol. 39, No. 5, 2002, pp. 1749–1779. https://doi.org/10.1137/S0036142901384162.

[36] Oliver, T. A., and Darmofal, D. L., "Analysis of dual consistency for discontinuous Galerkin discretizations of source terms," *SIAM Journal on Numerical Analysis*, Vol. 47, No. 5, 2009, pp. 3507–3525. https://doi.org/10.1137/080721467.

[37] Sleeman, M., "Goal-oriented model reduction for time-dependent nonlinear parametrized partial differential equations," Master's thesis, University of Toronto, 2020.

[38] Sleeman, M., and Yano, M., "Goal-oriented model reduction for parametrized time-dependent nonlinear partial differential equations," *submitted*, 2021.

[39] Haasdonk, B., and Ohlberger, M., "Reduced basis method for finite volume approximations of parametrized linear evolution equations," *Mathematical Modelling and Numerical Analysis*, Vol. 42, No. 2, 2008, pp. 277–302. https://doi.org/10.1051/m2an:2008001.

[40] Alexander, R., "Diagonally implicit Runge-Kutta methods for stiff O.D.E.'s," *SIAM Journal on Numerical Analysis*, Vol. 14, No. 6, 1977, pp. 1006–1021. https://doi.org/10.1137/0714068.

[41] Du, E., "A model reduction framework with the empirical quadrature procedure for high-dimensional shape-parameterized partial differential equation," Master's thesis, University of Toronto, 2020.

[42] Hesthaven, J. S., Stamm, B., and Zhang, S., "Efficient greedy algorithms for high-dimensional parameter spaces with applications to empirical interpolation and reduced basis methods," *ESAIM: M2AN*, Vol. 48, No. 1, 2014, pp. 259–283. https://doi.org/10.1051/m2an/2013100, URL https://doi.org/10.1051/m2an/2013100.

[43] Spalart, P. R., and Allmaras, S. R., "A one-equation turbulence model for aerodynamics flows," *La Recherche Aérospatiale*, Vol. 1, 1994, pp. 5–21.

[44] Allmaras, S. R., Johnson, F. T., and Spalart, P. R., "Modifications and clarifications for the implementation of the Spalart-Allmaras turbulence model," 7th International Conference on Computational Fluid Dynamics ICCFD7-1902, ICCFD, 2012.

[45] Schaefer, J., Cary, A., Mani, M., and Spalart, P., "Uncertainty quantification and sensitivity analysis of SA turbulence model coefficients in two and three dimensions," AIAA 2017-1710, AIAA, 2017. https://doi.org/10.2514/6.2017-1710.

[46] Barth, T. J., "An overview of combined uncertainty and a posteriori error bound estimates for CFD calculations," AIAA 2016-1062, AIAA, 2016.

[47] Donoghue, G., and Yano, M., "Spatio-stochastic adaptive discontinuous Galerkin methods," *Computer Methods in Applied Mechanics and Engineering*, Vol. 374, 2021, p. 113570. https://doi.org/10.1016/j.cma.2020.113570.

[48] Ohlberger, M., and Rave, S., "Nonlinear reduced basis approximation of parameterized evolution equations via the method of freezing," *Comptes Rendus Mathematique*, Vol. 351, No. 23-24, 2013, pp. 901–906. https://doi.org/10.1016/j.crma.2013.10.028.

[49] Nair, N. J., and Balajewicz, M., "Transported snapshot model order reduction approach for parametric, steady-state fluid flows containing parameter-dependent shocks," *International Journal for Numerical Methods in Engineering*, Vol. 117, No. 12, 2019, pp. 1234–1262. https://doi.org/10.1002/nme.5998.