Received: Added at production | Revised: Added at production | Accepted: Added at production

DOI: xxx/xxxx

ARTICLE TYPE

Goal-oriented model reduction of parametrized nonlinear PDEs; application to aerodynamics

Masayuki Yano

[1]Institute for Aerospace Studies, University of Toronto, Ontario, Canada

Correspondence
*Masayuki Yano, University of Toronto, 4925 Duffein Street, Toronto, ON, M3H 5T6, Canada. Email: myano@utias.utoronto.ca

Summary

We introduce a goal-oriented model reduction framework for rapid and reliable solution of parametrized nonlinear partial differential equations with applications in aerodynamics. Our goal is to provide quantitative and automatic control of various sources of errors in model reduction. Our framework builds on the following ingredients: a discontinuous Galerkin finite element (FE) method, which provides stability for convection-dominated problems; reduced basis (RB) spaces, which provide rapidly convergent approximations; the dual-weighted residual (DWR) method, which provides effective output error estimates for both the FE and RB approximations; output-based adaptive RB snapshots; and the empirical quadrature procedure (EQP), which hyperreduces the primal residual, adjoint residual, and output forms to enable online-efficient evaluations while providing quantitative control of hyperreduction errors. The framework constructs a reduced model which provides, for parameter values in the training set, output predictions that meet the user-prescribed tolerance by controlling the FE, RB, and EQP errors; in addition, the reduced model equips, for any parameter value, the output prediction with an effective, online-efficient error estimate. We demonstrate the framework for parametrized aerodynamics problems modeled by the Reynolds-averaged Navier-Stokes equations; reduced models provide over two orders of magnitude online computational reduction and sharp error estimates for three-dimensional flows.

KEYWORDS:
fluids, aerodynamics, model reduction, error estimation, hyperreduction, adaptivity

1 | INTRODUCTION

In this work we consider a goal-oriented framework for rapid and reliable solution of parametrized nonlinear partial differential equations (PDEs) with applications in aerodynamics. We are interested in the output prediction problem, in which we wish to evaluate an engineering quantity of interest — such as lift or drag — for a given configuration parameter — such as angle of attack and free stream Mach number. In particular, our interest is in many-query scenarios, in which the output must be evaluated for a large number of different input parameter values. Our approach to the problem is projection-based model reduction: in the offline stage, we construct, once, a reduced model through a relatively expensive exploration of the parameter space; in the online stage, we invoke, many times, the reduced model for many different parameter values. In this work, we introduce a goal-oriented model reduction framework for nonlinear PDEs which provides, in the offline stage, an efficient and automated training with quantitative output error control of various sources of model reduction errors and, in the online stage, rapid output predictions and associated error estimates.

To make the goal-oriented model reduction setting mathematically precise, we define an abstract form of the output prediction problem considered throughout this work. We introduce a parameter space $\mathcal{D} \subset \mathbb{R}^P$, a spatial domain $\Omega \subset \mathbb{R}^d$, a Hilbert space $\mathcal{V}$ over Ω, a residual form $r : \mathcal{V} \times \mathcal{V} \times \mathcal{D} \to \mathbb{R}$, and an output functional $q : \mathcal{V} \times \mathcal{D} \to \mathbb{R}$. The output prediction problem is as follows: given a parameter $\mu \in \mathcal{D}$, find the state $u(\mu) \in \mathcal{D}$ such that $r(u(\mu), v; \mu) = 0 \ \forall v \in \mathcal{V}$ and evaluate the output $s(\mu) = q(u(\mu); \mu) \in \mathbb{R}$. Many engineering problems are of this form, where the parameter $\mu \in \mathcal{D}$ is mapped to the state $u(\mu) \in \mathcal{V}$, which in turn is mapped to the output $s(\mu) \in \mathbb{R}$. For instance, in flight-envelope characterization in aerodynamics, the parameter may be the angle of attack and Mach number, the state is the flow field, and the output may be lift, drag, or moment.

For one- or few-query scenarios, we may consider a finite element (FE) approximation of the output prediction problem. To this end, we introduce a FE approximation space $\mathcal{V}_h$ of dimension $\mathcal{N}$, FE residual form $r_h : \mathcal{V}_h \times \mathcal{V}_h \times \mathcal{D} \to \mathbb{R}$, and FE output functional $q_h : \mathcal{V}_h \times \mathcal{D} \to \mathbb{R}$. We then consider the FE problem: given $\mu \in \mathcal{D}$, find $u_h(\mu) \in \mathcal{V}_h$ such that $r_h(u_h(\mu), v; \mu) = 0$ $\forall v \in \mathcal{V}_h$ and evaluate $s_h(\mu) = q_h(u_h(\mu); \mu)$. This work employs a discontinuous Galerkin (DG) method, which provides stability for conservation laws and is high-order accurate; we refer to[1,2] for reviews of DG methods. The solution of the FE problem requires $\mathcal{O}(\mathcal{N}^\beta)$ for some power β greater than 1, which may be fast enough for few-query scenarios but not for many-query scenarios.

For many-query and real-time scenarios, we consider the reduced basis (RB) approximation[3,4]. To this end, we introduce an RB space $\mathcal{V}_N \equiv \mathrm{span}\{u_h(\mu)\}_{\mu \in \Xi_N^{\mathrm{rb}}} \subset \mathcal{V}_h$ of dimension $N \ll \mathcal{N}$ associated with the snapshot parameter set $\Xi_N^{\mathrm{rb}} \subset \mathcal{D}$; the RB space is designed to approximate the parametric manifold $\{u_h(\mu) \mid \mu \in \mathcal{D}\}$. We then consider the RB problem: given $\mu \in \mathcal{D}$, find $u_N(\mu) \in \mathcal{V}_N$ such that $r_h(u_N(\mu), v; \mu) = 0 \ \forall v \in \mathcal{V}_N$ and evaluate $s_N(\mu) = q_h(u_N(\mu); \mu)$. If the PDE is linear and admits a so-called affine decomposition, the offline-online computational decomposition of the problem is straightforward; in the online stage, the reduced model can provide an output prediction $\mu \mapsto s(\mu)$ *and* an output error bound for $|s_h(\mu) - s_N(\mu)|$ in $\mathcal{O}(N^\gamma)$ operations, for some power γ greater than 1. We refer to a review paper[3] for a (goal-oriented) RB treatment of linear problems.

The goal of this work is to extend the goal-oriented RB formulation to PDEs with general nonlinearity, general parametric dependence, and a wide range of spatial scales. In the next three paragraphs, we discuss our approach to (1) provide online-efficient output predictions, (2) provide online-efficient error estimates, and (3) enable automatic construction of accurate reduced models. All the ingredients of our approach is developed with the aim to bring to complex nonlinear problems the similar level of automatic and quantitative error control that the RB method achieves for linear problems; we wish to minimize user intervention and case-specific tuning.

To provide online-efficient output predictions for nonlinear PDEs, we consider so-called hyperreduction of the reduced model (Section 2). We introduce a hyperreduced residual form $\tilde{r}_h : \mathcal{V}_h \times \mathcal{V}_h \times \mathcal{D} \to \mathbb{R}$ and an output functional $\tilde{q}_h : \mathcal{V}_h \times \mathcal{D} \to \mathbb{R}$, which (i) can be evaluated in $\mathcal{O}(N)$ operations and (ii) approximate the FE forms in the sense that $\tilde{r}_h(w, v; \mu) \approx r_h(w, v; \mu) \, \forall w, v \in \mathcal{V}_N$ and $\tilde{q}_h(w; \mu) \approx q_h(w; \mu) \ \forall w \in \mathcal{V}_N$. We then consider the hyperreduced RB problem: given $\mu \in \mathcal{D}$, find $\tilde{u}_N(\mu) \in \mathcal{V}_N$ such that $\tilde{r}_h(\tilde{u}_N(\mu), v; \mu) = 0 \ \forall v \in \mathcal{V}_N$ and evaluate $\tilde{s}_N(\mu) = \tilde{q}_h(\tilde{u}_N(\mu); \mu)$. Since $\dim(\mathcal{V}_N) = N$ and the evaluation of $\tilde{r}_h(\cdot, \cdot; \cdot)$ and $\tilde{q}_h(\cdot; \cdot)$ requires $\mathcal{O}(N)$ operations, the solution of the hyperreduced RB problem requires $\mathcal{O}(N^\gamma)$ operations; the approximation is hence online efficient. There exist many hyperreduction approaches, including the empirical interpolation method (EIM)[5], *the* hyperreduction method[6], the optimized cubature method[7], Gauss-Newton approximate tensor (GNAT) method[8], and the energy-conserving sampling and weighting (ECSW) method[9], to name a few. In this work we employ the empirical quadrature procedure (EQP)[10,11], which provides (i) a sparse quadrature for $\tilde{r}_h(\cdot, \cdot; \cdot)$ and $\tilde{q}_h(\cdot; \cdot)$ to enable $\mathcal{O}(N)$ evaluation and (ii) quantitative control of the solution error $\|u_N(\mu) - \tilde{u}_N(\mu)\|_{\mathcal{V}}$ due to hyperreduction; in this work we extend the EQP to the goal-oriented setting to control the output error $|s(\mu) - \tilde{s}_N(\mu)|$ due to hyperreduction.

To provide online-efficient error estimates for nonlinear PDEs, we apply the dual-weighted residual (DWR) method to the RB(-EQP) problem (Section 3). The original DWR method was proposed by Becker and Rannacher[12] to estimate the FE output error $|s(\mu) - s_h(\mu)|$ in $\mathcal{O}(\mathcal{N}^\beta)$ operations using the residual and adjoint $z_{\hat{h}}(\mu)$ computed in an enriched FE space $\mathcal{V}_{\hat{h}} \supset \mathcal{V}_h$. The DWR method, while not providing error *bounds*, provides effective error *estimates* for hyperbolic and convection-dominated problems and has been used successfully in many aerodynamics applications[13]. The DWR method has been used also in the context of model reduction for offline training[14] and for online adaptation (but without hyperreduction)[15]. In this work we extend the framework to the RB-EQP formulation: we first approximate the adjoint in an adjoint RB space $\mathrm{span}\{z_h(\mu)\}_{\mu \in \Xi_N^{\mathrm{rb}}}$; we then appeal to the DWR method to construct an output error estimate $\eta_h^{\mathrm{rb}}(\mu)$ for $|s_h(\mu) - s_N(\mu)|$. To treat nonlinearity, we again appeal to the EQP to (i) evaluate the adjoint and residual in $\mathcal{O}(N^\gamma)$ operations and (ii) provide quantitative control of the error in the error estimate $|\eta_N^{\mathrm{rb}}(\mu) - \tilde{\eta}_N^{\mathrm{rb}}(\mu)|$ due to hyperreduction. Our RB-EQP formulation for the DWR does not provide rigorous error *bounds* like the standard RB formulation for linear problems[3] or the Brezzi-Rappaz-Raviart formulation for quadratic nonlinearity[16]; it merely provides error *estimates*. However, it readily provides error estimates for hyperbolic and convection-dominated PDEs with general nonlinearity and parametric dependence. In addition, the DWR approach eliminates the need for a stability constant (i.e., inf-sup constant), and inasmuch also provides much sharper error estimates for convection-dominated PDEs.

To enable automatic construction of accurate reduced models for nonlinear PDEs that exhibit a wide range of scales, we incorporate in the simultaneous RB and EQP training algorithm proposed in[11] a spatially adaptive computation of the snapshots (Section 4). The use of spatially adaptive RB snapshots has been considered for linear PDEs in, e.g.,[17,18]. In particular, in[18] we devised, for linear coercive PDEs, a spatio-parameter adaptive offline training algorithm which constructs a reduced model with an output error bound on $|s(\mu) - s_N(\mu)|$ for all μ in the training set $\Xi_J^{\mathrm{train}} \supset \Xi_N^{\mathrm{rb}}$. As the construction of output error bounds is infeasible for general nonlinear PDEs, we settle for estimation and control of $|s(\mu) - \tilde{s}_N(\mu)|$ for snapshot parameters $\mu \in \Xi_N^{\mathrm{rb}}$, and the estimation and control of $|s_h(\mu) - \tilde{s}_N(\mu)|$ for $\mu \in \Xi_J^{\mathrm{train}} \setminus \Xi_N^{\mathrm{rb}}$. The spatio-parameter adaptivity is almost a necessity for accurate prediction of high Reynolds number aerodynamics[13], and we demonstrate its efficacy in Section 5.

We demonstrate our goal-oriented model reduction framework for two- and three-dimensional turbulent aerodynamics problems (Section 5). The previous works on model reduction of parametrized *and* nonlinear aerodynamics flows include works by LeGresley and Alonso[19,20,21], Washabaugh et al[22], and Zimmermann and Görtz[23,24]. (Given the application focus of this work, we here omit works on nonparametrized or linearized problems.) The key differentiator of the proposed goal-oriented formulation is that it enables "automated" construction of aerodynamic reduced models that meets the user-prescribed output error tolerance by quantitatively and automatically controlling the FE, RB, and EQP hyperreduction errors.

We summarize the fivefold contributions of this work. First, we introduce an EQP which provides quantitative control of the output error $|s_N(\mu) - \tilde{s}_N(\mu)|$ due to hyperreduction of the residual and output forms. Second, we introduce an online-efficient RB output error estimate $\tilde{\eta}_N^{\mathrm{rb}}(\mu)$ for $|s_h(\mu) - \tilde{s}_N(\mu)|$ based on the DWR error estimate; an EQP controls the error in the output error estimate $|\eta_N^{\mathrm{rb}}(\mu) - \tilde{\eta}_N^{\mathrm{rb}}(\mu)|$ due to hyperreduction of the residual and the adjoint problem. Third, we extend the simultaneous RB and EQP training procedure to the context of output prediction *and* error estimation. Fourth, we incorporate output-based adaptive snapshot calculation to control $|s(\mu) - s_h(\mu)|$ for $\mu \in \Xi_N^{\mathrm{rb}}$. Fifth, we demonstrate the goal-oriented model reduction framework for parametrized aerodynamics flows over an RAE 2822 airfoil and an ONERA M6 wing governed by the Reynolds-averaged Navier-Stokes equations; the framework constructs, in a fully automated manner, reduced models that provide over two orders of magnitude online computational reduction and sharp error estimates for the three-dimensional flows.

2 | OUTPUT PREDICTION

In this section we present our approach to the output prediction problem. Specifically, we introduce notations in Section 2.1, a precise form of the output prediction problem in Section 2.2, its FE approximation in Section 2.3, its RB approximation without hyperreduction in Section 2.4, and its RB-EQP approximation in Sections 2.5–2.7.

2.1 | Notations

To concisely describe our formulation for systems of PDEs, we adhere to the standard vector and tensor notations in this work. Given vectors (i.e., order-1 tensors) $w \in \mathbb{R}^m$ and $v \in \mathbb{R}^m$, their dot product is given by $w \cdot v = \sum_{i=1}^m w_i v_i \in \mathbb{R}$, and their Hadamard (i.e., element-wise) product is $w \circ v \in \mathbb{R}^m$ whose i-th entry is $w_i v_i$. Given vectors $w \in \mathbb{R}^m$ and $v \in \mathbb{R}^n$, their outer product is $w \otimes v \in \mathbb{R}^{m \times n}$ whose (i, j) entry is $w_i v_j$. Given matrices (i.e., order-2 tensors) $W \in \mathbb{R}^{m \times n}$ and $V \in \mathbb{R}^{m \times n}$, their double dot product is given by $W : V = \sum_{i=1}^m \sum_{j=1}^n W_{ij} V_{ij} \in \mathbb{R}$. Given an order-4 tensor $K \in \mathbb{R}^{m \times n \times m \times n}$ and an order-2 tensor $V \in \mathbb{R}^{m \times n}$, their product is $KV \in \mathbb{R}^{m \times n}$ whose (i, j) entry is given by $(KV)_{i,j} = \sum_{k=1}^m \sum_{l=1}^n K_{ijkl} V_{kl}$.

We in addition introduce notations associated with differentiable functions. Given a vector-valued function $v \in H^1(\Omega)^m$ over $\Omega \subset \mathbb{R}^d$, its gradient $\nabla v \in L^2(\Omega)^{m \times d}$ is matrix-valued and is given by $(\nabla v)_{ij} = \frac{\partial v_i}{\partial x_j}$ for $i = 1, \dots, m$ and $j = 1, \dots, d$. Given a matrix-valued function $V \in H^1(\Omega)^{m \times d}$, its divergence $\nabla \cdot V \in L^2(\Omega)^m$ is vector-valued and is given by $(\nabla \cdot V)_i = \sum_{j=1}^d \frac{\partial V_{ij}}{\partial x_j}$ for $i = 1, \dots, m$. Similarly, given a matrix-valued function $V \in H^1(\Omega)^{m \times d}$ and a normal vector n on $\partial\Omega$, their dot product $n \cdot V$ is vector-valued and is given by $(n \cdot V)_i = \sum_{j=1}^d n_j V_{ij}$ for $i = 1, \dots, m$. Note that both the divergence and the dot product with a normal vector contracts the second index of the matrix-valued function.

2.2 | Problem statement

We first define the precise form of our output prediction problem governed by a system of second-order nonlinear PDEs. To this end, we introduce a parameter space $\mathcal{D} \subset \mathbb{R}^P$ and a spatial domain $\Omega \subset \mathbb{R}^d$ with a Lipschitz boundary $\partial\Omega$. We then consider the following problem: given a parameter $\mu \in \mathcal{D}$, find the state $u(\mu)$ that satisfies

$$\nabla \cdot F(u(\mu);\mu) - \nabla \cdot K(u(\mu);\mu)\nabla u(\mu) = S(u(\mu), \nabla u(\mu);\mu) \quad \text{in } \Omega$$

and appropriate boundary conditions, and then evaluate the output

$$s(\mu) \equiv q(u(\mu);\mu) \equiv \int\limits_{\Omega} q^{\mathrm{v}}(u(\mu);\mu)dx + \int\limits_{\partial\Omega} q^{\mathrm{b}}(u(\mu), n \cdot K(u(\mu);\mu)\nabla u(\mu);\mu)ds;$$

here $F(\cdot;\cdot)$ is the advection flux, $K(\cdot;\cdot)$ is the diffusion tensor, $S(\cdot,\cdot;\cdot)$ is the source function, and $q(\cdot;\cdot)$ is the output functional which is defined by the volume output integrand $q^{\mathrm{v}}(\cdot;\cdot)$ and the boundary output integrand $q^{\mathrm{b}}(\cdot,\cdot;\cdot)$. Note that we consider second-order PDEs whose viscous term is linear in the gradient because (i) all aerodynamics equations of our interest are in this form and (ii) the form admits various discontinuous Galerkin discretizations. Throughout this work, we refer to this problem as the output prediction problem.

2.3 | Discontinuous Galerkin (DG) method

We now review the DG method based on an upwinded numerical flux and Bassi and Rebay's second discretization (i.e., BR2)[25], whose complete form in the presence of gradient-dependent source term is presented in[26]. We in particular consider an element-wise decomposition of the DG residual introduced in[27], which yields hyperreduced residuals that are stable for linear advection and advection-diffusion systems. We here describe the DG formulation for completeness, and refer to review papers[1,2] and textbooks[28,29] for more detailed treatment.

We first introduce a tessellation $\mathcal{T}_h$ that comprises elements $\{\kappa\}$ such that $\kappa \cap \kappa' = \emptyset$ for $\kappa \neq \kappa'$ and $\cup_{\kappa \in \mathcal{T}_h} \bar{\kappa} = \bar{\Omega}$. We denote the set of all facets of $\mathcal{T}_h$ by $\Sigma_h \equiv \{\sigma\}$. The facet set Σ_h is divided into a boundary facet set $\Sigma_h^{\mathrm{b}} \equiv \{\sigma \in \Sigma_h \mid \sigma \cap \partial\Omega \neq \emptyset\}$ and an interior facet set $\Sigma_h^{\mathrm{i}} \equiv \Sigma_h \setminus \Sigma_h^{\mathrm{b}}$; an interior facet $\sigma \in \Sigma_h^{\mathrm{i}}$ has two abutting elements, whereas a boundary facet $\sigma \in \Sigma_h^{\mathrm{b}}$ has one abutting element. We then introduce a DG approximation space of discontinuous piecewise polynomials

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

where m is the number of components in the state, and p is the polynomial degree of the approximation space. The DG residual form $r_h : \mathcal{V}_h \times \mathcal{V}_h \times \mathcal{D} \to \mathbb{R}$ is given by

$$r_h(w, v;\mu) \equiv \sum_{\kappa \in \mathcal{T}_h} r_\kappa(w, v;\mu). \tag{1}$$

The elemental DG residual form $r_\kappa : \mathcal{V}_h \times \mathcal{V}_h \times \mathcal{D} \to \mathbb{R}$ can be further decomposed into

$$r_\kappa(w, v;\mu) \equiv r_\kappa^{\mathrm{c}}(w, v;\mu) + r_\kappa^{\mathrm{d}}(w, v;\mu) + r_\kappa^{\mathrm{s}}(w, v;\mu),$$

where the superscripts c, d, and s denote the convection, diffusion, and source contributions, respectively.

The elemental DG residual form associated with the convection term is given by

$$\begin{aligned} r_\kappa^{\mathrm{c}}(w, v;\mu) \equiv &- \int\limits_{\kappa} \nabla v : F(w;\mu)dx + \int\limits_{\partial\kappa \cap \Sigma^{\mathrm{i}}} \left(\frac{1}{4}[(v \otimes n) : F(w;\mu)]_-^+ + \frac{1}{2}[v]_-^+ \cdot \hat{F}(w^+, w^-; n^+;\mu)\right) ds \\ &+ \int\limits_{\partial\kappa \cap \Sigma^{\mathrm{b}}} v^+ \cdot \hat{F}^{\mathrm{b}}(w^+; n^+;\mu)ds; \end{aligned}$$

here w^+ is w evaluated on $\partial\kappa$ from the side of κ, w^- is w evaluated on $\partial\kappa$ from the side of the neighboring element, $F : \mathbb{R}^m \times \mathcal{D} \to \mathbb{R}^{m \times d}$ is the convection flux function, $\hat{F} : \mathbb{R}^m \times \mathbb{R}^m \times \mathbb{R}^d \times \mathcal{D} \to \mathbb{R}^m$ is an interior numerical flux function, and $\hat{F}^{\mathrm{b}} : \mathbb{R}^m \times \mathbb{R}^d \times \mathcal{D} \to \mathbb{R}^m$ is a boundary numerical flux function. We note that the non-standard term on the interior facets, $[(v \otimes n) : F(w;\mu)]_-^+$, is required to ensure that the hyperreduced system is energy stable for linear hyperbolic systems[27].

The elemental DG residual form associated with the diffusion term is given by

$$\begin{aligned} r_\kappa^{\mathrm{d}}(w, v; \mu) \equiv & \int_\kappa \nabla v : K(w;\mu) \nabla w dx \\ & - \int_\sigma \left(\frac{1}{2} \nabla v^+ : K(w^+;\mu) \llbracket w \rrbracket + \frac{1}{2} \llbracket v \rrbracket : K(w^+;\mu)(\nabla w^+ + \theta_\sigma \ell_\sigma^+(\llbracket w \rrbracket)) \right) ds \\ & - \int_\sigma \Big(\nabla v : K(w^+;\mu)((w^+ - u^b(w^+;\mu)) \otimes n) \\ & \qquad + (v^+ \otimes n^+) : \left(K(w^+;\mu)(\nabla w^+ + \theta_\sigma \ell_\sigma^{\mathrm{b}}(w^+;\mu)) \right) \Big) ds, \end{aligned}$$

where $\llbracket w \rrbracket \equiv w^+ \otimes n^+ + w^- \otimes n^-$ is the jump operator, $K : \mathbb{R}^m \times \mathcal{D} \to \mathbb{R}^{m \times d \times m \times d}$ is the diffusion tensor, $u^{\mathrm{b}} : \mathbb{R}^m \times \mathcal{D} \to \mathbb{R}^m$ is the boundary state function, ℓ_σ^+ is the interior lifting operator, ℓ_σ^{b} is the boundary lifting operator, and θ_σ is the lifting operator penalty factor. We refer to[25] for the lifting operators for the BR2 scheme and[2] for other choices of lifting operators. This particular element-wise decomposition of the residual ensures that, for linear equations, each elemental matrix associated with the diffusion term is symmetric positive semi-definite and hence the hyperreduced system is energy-stable for any choice of non-negative weights[27].

The elemental DG residual form associated with the source term is given by

$$r_\kappa^{\mathrm{s}}(w, v; \mu) \equiv \int_\kappa v \cdot S(w, \nabla w + \ell_\kappa^{\mathrm{e}}(w); \mu) dx$$

where $S : \mathbb{R}^m \times \mathcal{D} \to \mathbb{R}^m$ is the source function, and ℓ_κ^{e} is the element lifting operator given by

$$\ell_\kappa^{\mathrm{e}}(w) \equiv \sum_{\sigma \in \partial\kappa \cap \Sigma^{\mathrm{i}}} \ell_\sigma^+(\llbracket w \rrbracket) + \sum_{\sigma \in \partial\kappa \cap \Sigma^{\mathrm{b}}} \ell_\sigma^{\mathrm{b}}(w^+).$$

The lifting of the gradient is required to ensure that the DG residual is asymptotically dual consistent, a required condition for output superconvergence[30].

We also introduce an element-wise decomposition of the output functional. The DG output functional $q_h : \mathcal{V}_h \times \mathcal{D} \to \mathbb{R}$ is given by

$$q_h(w;\mu) \equiv \sum_{\kappa \in \mathcal{T}_h} q_\kappa(w;\mu). \tag{2}$$

The elemental DG output form $q_\kappa : \mathcal{V}_h \times \mathcal{D} \to \mathbb{R}$ is given by

$$q_\kappa(w;\mu) \equiv \int_\kappa q^{\mathrm{v}}(w;\mu) dx + \int_{\partial\kappa \cap \Sigma^{\mathrm{b}}} q^{\mathrm{b}}(w^+, n \cdot K(w^+;\mu)(\nabla w^+ + \theta_\sigma \ell_\sigma^{\mathrm{b}}(w^+;\mu)); \mu) ds,$$

where $q^{\mathrm{v}} : \mathbb{R}^m \times \mathcal{D} \to \mathbb{R}$ is the volume output function, $q^{\mathrm{b}} : \mathbb{R}^m \times \mathbb{R}^m \times \mathcal{D} \to \mathbb{R}$ is the boundary output function, and the superscript v and b denote the volume and boundary outputs, respectively. We note that the boundary output is a function of the boundary state and the diffusion flux based on the lifted gradient.

Having defined the DG residual and output forms, we now introduce the DG-FEM, or more simply FE, output prediction problem: given $\mu \in \mathcal{D}$, find $u_h(\mu) \in \mathcal{V}_h$ such that

$$r_h(u_h(\mu), v; \mu) = 0 \quad \forall v \in \mathcal{V}_h, \tag{3}$$

and evaluate the output

$$s_h(\mu) \equiv q_h(u_h(\mu); \mu),$$

where $r_h(\cdot, \cdot, ; \cdot)$ and $q_h(\cdot; \cdot)$ are the DG residual form (1) and output form (2), respectively. We assume that the FE problem (3) is well posed.

2.4 | RB output prediction

We now consider an RB approximation of the output prediction problem. To this end, we first introduce an RB parameter set $\Xi_N^{\mathrm{rb}} = \{\mu^{(i)}\}_{i=1}^N$ that comprises N snapshot parameter values; a systematic procedure to select the parameter set is discussed in Section 4.2. We next introduce the associated (hierarchical, primal) RB spaces $\mathcal{V}_N^{\mathrm{pr}} \equiv \operatorname{span}\{u_h(\mu)\}_{\mu \in \Xi_N^{\mathrm{rb}}} \subset \mathcal{V}_h$ for $N =$

$1, \ldots, N_{\max} \leq \dim(\mathcal{V}_h)$; in practice, $N_{\max} \ll \dim(\mathcal{V}_h)$. We then introduce the RB output prediction problem: given $\mu \in \mathcal{D}$, find $u_N(\mu) \in \mathcal{V}_N^{\mathrm{pr}}$ such that

$$r_h(u_N(\mu), v; \mu) = 0 \quad \forall v \in \mathcal{V}_N^{\mathrm{pr}}, \tag{4}$$

and evaluate the output

$$s_N(\mu) \equiv q_h(u_N(\mu); \mu),$$

where $r_h(\cdot, \cdot, ; \cdot)$ and $q_h(\cdot; \cdot)$ are the DG residual form (1) and output form (2), respectively. We assume that the RB problem (4) is well posed.

For notational convenience, we also introduce an algebraic form of the problem. To this end, we first introduce a $\mathcal{V}$-orthonormal basis $\{\phi_i^{\mathrm{pr}}\}_{i=1}^N$ of $\mathcal{V}_N^{\mathrm{pr}}$. We next introduce an operator $V_N^{\mathrm{pr}} : \mathbb{R}^N \to \mathcal{V}_N^{\mathrm{pr}}$ which associates a generalized coordinate $\mathbf{w} \in \mathbb{R}^N$ to a function $w = V_N^{\mathrm{pr}}\mathbf{w} = \sum_{j=1}^N \phi_j \mathbf{w}_j \in \mathcal{V}_N^{\mathrm{pr}}$. We then introduce the algebraic RB residual evaluated with respect to primal RB and output functional: $\mathbf{r}_N^{\mathrm{pr}} : \mathbb{R}^N \times \mathcal{D} \to \mathbb{R}^N$ and $\mathbf{q}_N : \mathbb{R}^N \times \mathcal{D} \to \mathbb{R}$ such that

$$\begin{aligned} \mathbf{r}_N^{\mathrm{pr}}(\mathbf{w}; \mu)_i &\equiv r_h(V_N^{\mathrm{pr}}\mathbf{w}, \phi_i^{\mathrm{pr}}; \mu), \quad i = 1, \ldots, N, \\ \mathbf{q}_N(\mathbf{w}; \mu) &\equiv q_h(V_N^{\mathrm{pr}}\mathbf{w}; \mu). \end{aligned}$$

The RB primal problem is as follows: given $\mu \in \mathcal{D}$, find $\mathbf{u}_N(\mu) \in \mathbb{R}^N$ such that

$$\mathbf{r}_N^{\mathrm{pr}}(\mathbf{u}_N(\mu); \mu) = 0 \quad \text{in } \mathbb{R}^N, \tag{5}$$

and evaluate the output

$$s_N(\mu) \equiv \mathbf{q}_N(\mathbf{u}_N(\mu); \mu).$$

Given the RB problem (4) is well posed, the algebraic RB problem (5) is well posed and $u_N(\mu) = V_N^{\mathrm{pr}}\mathbf{u}_N(\mu)$.

Remark 1. The RB approximation of the output often includes the so-called dual correction term and takes on the form $s_N(\mu) \equiv q_h(u_N(\mu); \mu) - r_h(u_N(\mu), z(\mu); \mu)$, where $z(\mu)$ is an approximation to the dual solution computed in a dual RB space that differs from $\mathcal{V}_N^{\mathrm{pr}}$[3]. However, in this work we refrain from including the dual correction in the output prediction itself, and instead use the correction term to estimate the error in the output.

2.5 | RB-EQP output prediction

We now introduce the hyperreduced residual and output forms associated with an RB-EQP approximation. Throughout this work, we denote the EQP approximation of an operator (e.g., residual) or a variable associated with the EQP approximation (e.g., solution) with a tilde $\tilde{(\cdot)}$. We first introduce the EQP residual associated with a given set of residual EQP weights $\{\rho_\kappa^r \in \mathbb{R}_{\geq 0}\}_{\kappa \in \mathcal{T}_h}$:

$$\tilde{r}_h(w, v; \mu) \equiv \sum_{\kappa \in \mathcal{T}_h} \rho_\kappa^r r_\kappa(w, v; \mu); \tag{6}$$

the superscript r on ρ_κ^r signifies the EQP weight is associated with the residual form $r_\kappa(\cdot, \cdot; \cdot)$. We next introduce the EQP output functional associated with a given set of output-functional EQP weights $\{\rho_\kappa^q \in \mathbb{R}_{\geq 0}\}_{\kappa \in \mathcal{T}_h}$:

$$\tilde{q}_h(w; \mu) \equiv \sum_{\kappa \in \mathcal{T}_h} \rho_\kappa^q q_\kappa(w; \mu); \tag{7}$$

the superscript q on ρ_κ^q signifies the EQP weight is associated with the output functional $q_\kappa(\cdot; \cdot)$. The EQP weights are found using a linear programming (LP) procedure described shortly. We then introduce the RB-EQP output prediction problem: given $\mu \in \mathcal{D}$, find $\tilde{u}_N(\mu) \in \mathcal{V}_N^{\mathrm{pr}}$ such that

$$\tilde{r}_h(\tilde{u}_N(\mu), v; \mu) = 0 \quad \forall v \in \mathcal{V}_N^{\mathrm{pr}} \tag{8}$$

and evaluate the output

$$\tilde{s}_N(\mu) \equiv \tilde{q}_h(\tilde{u}_N(\mu); \mu). \tag{9}$$

We assume the RB-EQP problem is well posed.

We can again recast the problem in an algebraic form. To this end, we introduce the discrete RB-EQP residual and output functional evaluated with respect to the primal RB: $\tilde{\mathbf{r}}_N^{\mathrm{pr}} : \mathbb{R}^N \times \mathcal{D} \to \mathbb{R}^N$ and $\tilde{\mathbf{q}}_N : \mathbb{R}^N \times \mathcal{D} \to \mathbb{R}$ such that

$$\begin{aligned} \tilde{\mathbf{r}}_N^{\mathrm{pr}}(\mathbf{w}; \mu)_i &\equiv \tilde{r}_h(V_N^{\mathrm{pr}}\mathbf{w}, \phi_i^{\mathrm{pr}}; \mu), \quad i = 1, \ldots, N, \\ \tilde{\mathbf{q}}_N(\mathbf{w}; \mu) &\equiv \tilde{q}_h(V_N^{\mathrm{pr}}\mathbf{w}; \mu). \end{aligned}$$

The RB primal problem is as follows: given $\mu \in \mathcal{D}$, find $\tilde{\mathbf{u}}_N(\mu) \in \mathbb{R}^N$ such that

$$\tilde{\mathbf{r}}_N^{\mathrm{pr}}(\tilde{\mathbf{u}}_N(\mu);\mu) = 0 \quad \text{in } \mathbb{R}^N,$$

and evaluate the output

$$\tilde{s}_N(\mu) \equiv \tilde{\mathbf{q}}_N(\tilde{\mathbf{u}}_N(\mu);\mu).$$

We note that $\tilde{u}_N(\mu) = V_N^{\mathrm{pr}} \tilde{\mathbf{u}}_N(\mu)$.

2.6 | EQP: general form of the linear program

We find the EQP weights $\{\rho_\kappa^r \in \mathbb{R}_{\geq 0}\}_{\kappa \in \mathcal{T}_h}$ and $\{\rho_\kappa^q \in \mathbb{R}_{\geq 0}\}_{\kappa \in \mathcal{T}_h}$ for the EQP residual form (6) and output form (7), respectively, using a linear programming (LP) procedure introduced for integration of parametric functions in [10] and for hyperreduction of RB residuals in [11]. The procedure considers ℓ_1 minimization of the EQP weights $\{\rho_\kappa^\bullet\}_{\kappa \in \mathcal{T}_h}$ subject to a set of constraints designed to control the error in the RB-EQP solution with respect to the RB solution; here the superscript $^\bullet$ may be r for the residual or q for the output functional (or later η for the error estimate). In [11], we introduced an LP procedure with a constraint that controls the $\mathcal{V}$-norm of the error in the RB-EQP solution due to hyperreduction, $\|u_N(\mu) - \tilde{u}_N(\mu)\|_{\mathcal{V}}$. As the goal of the current work is to control the hyperreduction error in the output and the associated error estimate, we will impose different LP constraints. To this end, we first introduce the general form of the LP, and introduce the specific constraints required for output error control in subsequent sections.

To introduce the LP procedure, we first introduce a parameter training set $\Xi_J^{\mathrm{train}} \equiv \{\hat{\mu}_j \in \mathcal{D}\}_{j=1}^J$ that comprises J training parameters. We then introduce the associated training state set $U_J^{\mathrm{train}} \equiv \{\hat{\mathbf{u}}_N(\mu) \in \mathbb{R}^N\}_{\mu \in \Xi_J^{\mathrm{train}}}$; the training state set in general need not be the set of RB states $\{\mathbf{u}_N(\mu)\}_{\mu \in \Xi_J^{\mathrm{train}}}$ or RB-EQP states $\{\tilde{\mathbf{u}}_N(\mu)\}_{\mu \in \Xi_J^{\mathrm{train}}}$. The general form of the LP, denoted $\mathrm{LP}^\bullet(\Xi_J^{\mathrm{train}}, U_J^{\mathrm{train}}, \delta^\bullet)$, is as follows: find $\{\rho_\kappa^\bullet\}_{\kappa \in \mathcal{T}_h}$ such that

$$\{\rho_\kappa^\bullet\}_{\kappa \in \mathcal{T}_h} = \arg\min_{\{\rho_\kappa\}} \sum_{\kappa \in \mathcal{T}_h} \rho_\kappa$$

subject to non-negativity constraints

$$\rho_\kappa \geq 0 \quad \forall \kappa \in \mathcal{T}_h,$$

a constant integration constraint

$$\left| |\Omega| - \sum_{\kappa \in \mathcal{T}_h} \rho_\kappa |\kappa| \right| \leq \delta_\Omega,$$

and manifold accuracy constraints

$$|c(\{\rho_\kappa\}_{\kappa \in \mathcal{T}_h};\mu)|_i \leq \delta^\bullet, \quad i = 1, \dots, M, \ \forall \mu \in \Xi_J^{\mathrm{train}}; \tag{10}$$

here, $|\Omega|$ and $|\kappa|$ are the volumes of the domain Ω and element κ, respectively, and $c : \mathbb{R}^{|\mathcal{T}_h|} \times \mathcal{D} \to \mathbb{R}^M$ is linear in the first argument and imposes M manifold accuracy constraints per training set parameter. The particular value of the manifold accuracy tolerance $\delta^\bullet$ depends on the specific instance of the LP (e.g., the residual-form tolerance would be δ^r).

Remark 2. In practice, the M absolute-value manifold accuracy constraints for each $\mu \in \Xi_J^{\mathrm{train}}$ can be expressed as $2M$ linear inequality constraints in LP: i.e., $c(\{\rho_\kappa\}_{\kappa \in \mathcal{T}_h};\mu)_k \leq \delta^\bullet$ and $-c(\{\rho_\kappa\}_{\kappa \in \mathcal{T}_h};\mu)_k \leq \delta^\bullet$ for $k = 1, \dots, M$.

Remark 3. As we will see shortly, all of the manifold accuracy constraints $\mathbb{R}^{|\mathcal{T}_h|} \times \mathcal{D} \to \mathbb{R}^M$ that we consider in this work have a property that $c(\{\rho_\kappa = 1\}_{\kappa \in \mathcal{T}_h};\mu) = 0$ for all $\mu \in \mathcal{D}$; i.e., the exact (non-sparse) EQP weights satisfy the manifold accuracy constraints exactly. The weights $\{\rho_\kappa = 1\}_{\kappa \in \mathcal{T}_h}$ also satisfy the constant-integration constraints. As a result, the LP problem is guaranteed to have a feasible solution.

2.7 | EQP: output-prediction accuracy constraints

We now wish to identify the EQP manifold accuracy constraints to control the error in the RB-EQP output $\tilde{s}_N(\mu) = \tilde{\mathbf{q}}_N(\tilde{\mathbf{u}}_N(\mu);\mu)$ with respect to the RB output $s_N(\mu) = \mathbf{q}_N(\mathbf{u}_N(\mu);\mu)$. To this end, we note that the difference in the two outputs $s_N(\mu)$ and $\tilde{s}_N(\mu)$ is due to two sources of hyperreduction errors: (i) the EQP approximation of the residual $\mathbf{r}_N^{\mathrm{pr}}(\cdot;\mu)$, whose root is the RB-EQP solution $\tilde{\mathbf{u}}_N(\mu)$ which in general differs from the RB solution $\mathbf{u}_N(\mu)$; (ii) the EQP approximation of the output functional

$\mathbf{q}_N(\cdot;\mu)$ itself. Mathematically, we invoke the triangle inequality to obtain

$$\begin{aligned}|s_N(\mu)-\tilde{s}_N(\mu)| &= |\mathbf{q}_N(\mathbf{u}_N(\mu);\mu)-\tilde{\mathbf{q}}_N(\tilde{\mathbf{u}}_N(\mu);\mu)| \\ &\le |\mathbf{q}_N(\mathbf{u}_N(\mu);\mu)-\mathbf{q}_N(\tilde{\mathbf{u}}_N(\mu);\mu)| + |\mathbf{q}_N(\tilde{\mathbf{u}}_N(\mu);\mu)-\tilde{\mathbf{q}}_N(\tilde{\mathbf{u}}_N(\mu);\mu)|,\end{aligned}$$

where the first and second terms are associated with the sources (i) and (ii), respectively. We wish to control both sources of the error to provide accurate hyperreduced output predictions.

To control the error due to source (i) — the approximation of $\mathbf{r}_N^{\rm pr}(\cdot;\mu)$ —, we first introduce the dual problem *approximated in the primal RB space*: find $\mathbf{z}_N^{\rm pr}(\mu)\in\mathbb{R}^N$ such that

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

where

$$\begin{aligned}\mathbf{J}_N^{\rm pr}(\hat{\mathbf{u}}_N(\mu);\mu)_{ij} &\equiv r_h'(V_N^{\rm pr}\hat{\mathbf{u}}_N(\mu);\phi_j^{\rm pr},\phi_i^{\rm pr}), \quad i,j=1,\dots,N, \\ \mathbf{g}_N^{\rm pr}(\hat{\mathbf{u}}_N(\mu);\mu)_j &\equiv q_h'(V_N^{\rm pr}\hat{\mathbf{u}}_N(\mu);\phi_j^{\rm pr}), \quad j=1,\dots,N,\end{aligned}$$

where $r_h'(V_N^{\rm pr}\hat{\mathbf{u}}_N(\mu);\phi_j^{\rm pr},\phi_i^{\rm pr};\mu)$ and $q_h'(V_N^{\rm pr}\hat{\mathbf{u}}_N(\mu);\phi_j^{\rm pr};\mu)$ are the Gâteaux derivatives of $r_h(\cdot,\phi_i^{\rm pr};\mu)$ and $q_h(\cdot;\mu)$, respectively, about $V_N^{\rm pr}\hat{\mathbf{u}}_N(\mu)$ in the direction $\phi_j^{\rm pr}$. We next introduce a modified adjoint

$$\mathbf{z}_N^{\rm pr,w}(\mu) = \max\{|\mathbf{z}_N^{\rm pr}(\mu)|, z^{\rm pr,min}(\mu)\},$$

where $z^{\rm pr,min}(\mu) \equiv N^{1/2}\sqrt{\delta^r}\|\mathbf{z}_N^{\rm pr}(\mu)\|_2$ and the max operator between the vector and scalar is applied to each element of the vector; e.g., $\mathbf{z}_N^{\rm pr,w}(\mu)_i = \max\{|\mathbf{z}_N^{\rm pr}(\mu)|_i, z^{\rm pr,min}(\mu)\}$, $i=1,\dots,N$. This is the modified dual "weight" with a minimum value constraint and is denoted by the superscript "w". We then introduce the residual manifold accuracy constraints

$$|\mathbf{z}_N^{\rm pr,w}(\mu)\circ(\mathbf{r}_N^{\rm pr}(\hat{\mathbf{u}}(\mu);\mu)-\sum_{\kappa\in\mathcal{T}_h}\rho_\kappa^r\mathbf{r}_{N,\kappa}^{\rm pr}(\hat{\mathbf{u}}(\mu);\mu))|_i \le \frac{2\delta^r}{3N}, \quad i=1,\dots,N,\ \forall\mu\in\Xi_J^{\rm train}, \tag{11}$$

where $\circ$ denotes the Hadamard (i.e., element-wise) product as described in Section 2.1. Each of the N absolute-value constraint can be expressed as two linear inequality constraints as discussed in Remark 2, and hence the constraint is admissible in LP. In addition, the LP has a feasible solution for any $\Xi_J^{\rm train}\subset\mathcal{D}$, as the left hand side of the constraint evaluates to 0 for the exact (non-sparse) weights $\{\rho_\kappa^r=1\}_{\kappa\in\mathcal{T}_h}$. The LP procedure for the residual, denoted $\mathtt{LP}^r(\Xi_J^{\rm train}, U_J^{\rm train}\equiv\{\hat{\mathbf{u}}_N(\mu)\}_{\mu\in\Xi_J^{\rm train}},\delta^r)$, results from the substitution of the residual manifold accuracy constraint (11) in place of the generic manifold accuracy constraint (10) of the generic LP procedure $\mathtt{LP}^\bullet$. The number of manifold accuracy constraints per training parameter point is $M=N$; the total number of manifold accuracy constraints for J training parameter points is hence NJ (which can be written as $2NJ$ linear inequality constraints following Remark 2). We note that the manifold accuracy constraints are evaluated about a training state set $U_J^{\rm train}\equiv\{\hat{\mathbf{u}}_N(\mu)\}_{\mu\in\Xi_J^{\rm train}}$, which differs from $\{\tilde{\mathbf{u}}_N(\mu)\}_{\mu\in\Xi_J^{\rm train}}$; the use of the surrogate training set is necessary as the RB-EQP state $\tilde{\mathbf{u}}_N(\mu)$ depends on the RB-EQP residual $\tilde{\mathbf{r}}_N(\cdot;\mu)$, which itself depends on the solution to $\mathtt{LP}^r$.

To control the error due to source (ii) — the approximation of $\mathbf{q}_N(\cdot;\cdot)$ —, we introduce an output-functional manifold accuracy constraint

$$|\mathbf{q}_N(\tilde{\mathbf{u}}_N(\mu);\mu)-\sum_{\kappa\in\mathcal{T}_h}\rho_\kappa^q\mathbf{q}_{N,\kappa}(\tilde{\mathbf{u}}_N(\mu);\mu)| \le \delta^q \quad \forall\mu\in\Xi_J^{\rm train}. \tag{12}$$

We again note that each absolute-value constraint can be expressed as two linear inequality constraints and hence is admissible in LP; see Remark 2. In addition, the LP has a feasible solution for any $\Xi_J^{\rm train}\subset\mathcal{D}$, as the left hand side of the constraint evaluates to 0 for the exact (non-sparse) weights $\{\rho_\kappa^q=1\}_{\kappa\in\mathcal{T}_h}$. The LP procedure for the output functional, denoted $\mathtt{LP}^q(\Xi_J^{\rm train}, U_J^{\rm train}\equiv\{\tilde{\mathbf{u}}_N(\mu)\}_{\mu\in\Xi_J^{\rm train}},\delta^q)$, is obtained by substituting the output-functional manifold accuracy constraint (12) in place of the generic manifold accuracy constraint (10) of the generic LP procedure $\mathtt{LP}^\bullet$. The number of manifold accuracy constraints per training parameter point is $M=1$; the number of total constraints for J training parameter points is J (which can be written as $2J$ linear inequality constraints following Remark 2). Note that, in $\mathtt{LP}^r$, $\{\tilde{\mathbf{u}}_N(\mu)\}_{\mu\in\Xi_J^{\rm train}}$ are not yet known because $\tilde{\mathbf{u}}_N(\mu)$ depends on $\mathtt{LP}^r$; however, in $\mathtt{LP}^q$, the states $\{\tilde{\mathbf{u}}_N(\mu)\}_{\mu\in\Xi_J^{\rm train}}$ are already known. We hence use $U_J^{\rm train}\equiv\{\tilde{\mathbf{u}}_N(\mu)\}_{\mu\in\Xi_J^{\rm train}}$ in the EQP training of the output functional.

We readily observe that the constraint (12) directly controls the error due to source (ii) for $\mu\in\Xi_J^{\rm train}$. We now provide a proposition that relates the error due to source (i) to the constraint (11).

Proposition 1 (Output error due to EQP approximation of $\mathbf{r}_N^{\text{pr}}(\cdot;\cdot)$)**.** Suppose

$$\|\mathbf{z}_N^{\text{pr,w}} \circ (\mathbf{r}_N^{\text{pr}}(\hat{\mathbf{u}}) - \tilde{\mathbf{r}}_N^{\text{pr}}(\hat{\mathbf{u}}))\|_\infty \le \frac{2\delta^r}{3N}, \tag{13}$$

$$\|I - \mathbf{J}_N^{\text{pr}}(\hat{\mathbf{u}}_N)\tilde{\mathbf{J}}_N^{\text{pr}}(\hat{\mathbf{u}}_N)^{-1}\|_{\max} \le \delta_{\mathbf{J}}, \tag{14}$$

where $\|A\|_{\max}$ denotes the maximum entry of the matrix $A \in \mathbb{R}^{N\times N}$. Then

$$|\mathbf{q}_N(\mathbf{u}_N) - \mathbf{q}_N(\tilde{\mathbf{u}}_N)| \le \delta^r + \mathcal{O}(\delta_{\mathbf{J}}^2) + \mathcal{O}(\hat{\delta}^2) + \mathcal{O}(\tilde{\delta}^2),$$

where $\tilde{\delta} \equiv \|\mathbf{u}_N - \tilde{\mathbf{u}}_N\|_2$ and $\hat{\delta} \equiv \|\mathbf{u}_N - \hat{\mathbf{u}}_N\|_2$.

Proof. For notational convenience, we first define $\delta\hat{\mathbf{u}}_N \equiv \mathbf{u}_N - \hat{\mathbf{u}}_N$ and $\delta\tilde{\mathbf{u}}_N \equiv \mathbf{u}_N - \tilde{\mathbf{u}}_N$ so that $\hat{\delta} \equiv \|\delta\hat{\mathbf{u}}_N\|_2$ and $\tilde{\delta} \equiv \|\delta\tilde{\mathbf{u}}_N\|_2$. We then note that, by the Taylor series expansion about $\hat{\mathbf{u}}_N$,

$$\begin{aligned}\mathbf{q}_N(\mathbf{u}_N) - \mathbf{q}_N(\tilde{\mathbf{u}}_N) &= \mathbf{g}_N^{\text{pr}}(\hat{\mathbf{u}}_N)^T(\mathbf{u}_N - \hat{\mathbf{u}}_N) + \mathcal{O}(\|\mathbf{u}_N - \hat{\mathbf{u}}_N\|_2) - \mathbf{g}_N^{\text{pr}}(\hat{\mathbf{u}}_N)^T(\tilde{\mathbf{u}}_N - \hat{\mathbf{u}}_N) + \mathcal{O}(\|\tilde{\mathbf{u}}_N - \hat{\mathbf{u}}_N\|_2)\\ &= \mathbf{g}_N^{\text{pr}}(\hat{\mathbf{u}}_N)^T\delta\tilde{\mathbf{u}}_N + \mathcal{O}(\hat{\delta}^2) + \mathcal{O}(\tilde{\delta}^2) = \mathbf{z}_N^{\text{pr}\,T}\mathbf{J}_N^{\text{pr}}(\hat{\mathbf{u}}_N)\delta\tilde{\mathbf{u}}_N + \mathcal{O}(\hat{\delta}^2) + \mathcal{O}(\tilde{\delta}^2),\end{aligned} \tag{15}$$

where the second equality follows from the definitions of $\delta\tilde{\mathbf{u}}_N$ and $\delta\hat{\mathbf{u}}_N$ and the relationship $\|\tilde{\mathbf{u}}_N - \hat{\mathbf{u}}_N\|_2 = \|-\delta\tilde{\mathbf{u}}_N + \delta\hat{\mathbf{u}}_N\|_2 \le 2(\tilde{\delta}^2 + \hat{\delta}^2)$, and the last equality follows from the definition of the adjoint $\mathbf{z}_N^{\text{pr}}$. We also note that

$$\begin{aligned}&\mathbf{r}_N^{\text{pr}}(\hat{\mathbf{u}}_N) - \tilde{\mathbf{r}}_N^{\text{pr}}(\hat{\mathbf{u}}_N)\\ &= \mathbf{r}_N^{\text{pr}}(\mathbf{u}_N) - \mathbf{J}_N^{\text{pr}}(\mathbf{u}_N)\delta\hat{\mathbf{u}}_N - \tilde{\mathbf{r}}_N^{\text{pr}}(\tilde{\mathbf{u}}_N) - \tilde{\mathbf{J}}_N^{\text{pr}}(\tilde{\mathbf{u}}_N)(\delta\tilde{\mathbf{u}}_N - \delta\hat{\mathbf{u}}_N) + \mathcal{O}(\hat{\delta}^2) + \mathcal{O}(\tilde{\delta}^2)\\ &= -\mathbf{J}_N^{\text{pr}}(\hat{\mathbf{u}}_N)\delta\hat{\mathbf{u}}_N - \tilde{\mathbf{J}}_N^{\text{pr}}(\hat{\mathbf{u}}_N)(\delta\tilde{\mathbf{u}}_N - \delta\hat{\mathbf{u}}_N) + \mathcal{O}(\hat{\delta}^2) + \mathcal{O}(\tilde{\delta}^2)\\ &= -\tilde{\mathbf{J}}_N^{\text{pr}}(\hat{\mathbf{u}}_N)\delta\tilde{\mathbf{u}}_N + (\tilde{\mathbf{J}}_N^{\text{pr}}(\hat{\mathbf{u}}_N) - \mathbf{J}_N^{\text{pr}}(\hat{\mathbf{u}}_N))\delta\hat{\mathbf{u}}_N + \mathcal{O}(\hat{\delta}^2) + \mathcal{O}(\tilde{\delta}^2).\end{aligned}$$

It follows that

$$\delta\tilde{\mathbf{u}}_N = -\tilde{\mathbf{J}}_N^{\text{pr}}(\hat{\mathbf{u}}_N)^{-1}(\mathbf{r}_N^{\text{pr}}(\hat{\mathbf{u}}_N) - \tilde{\mathbf{r}}_N^{\text{pr}}(\hat{\mathbf{u}}_N)) + (I - \tilde{\mathbf{J}}_N^{\text{pr}}(\hat{\mathbf{u}}_N)^{-1}\mathbf{J}_N^{\text{pr}}(\hat{\mathbf{u}}_N))\delta\hat{\mathbf{u}}_N + \mathcal{O}(\hat{\delta}^2) + \mathcal{O}(\tilde{\delta}^2). \tag{16}$$

The substitution of (16) into (15) yields

$$\begin{aligned}&|\mathbf{q}_N(\mathbf{u}_N) - \mathbf{q}_N(\tilde{\mathbf{u}}_N)|\\ &= |-\mathbf{z}_N^{\text{pr}\,T}\mathbf{J}_N^{\text{pr}}(\hat{\mathbf{u}}_N)\tilde{\mathbf{J}}_N^{\text{pr}}(\hat{\mathbf{u}}_N)^{-1}(\mathbf{r}_N^{\text{pr}}(\hat{\mathbf{u}}_N) - \tilde{\mathbf{r}}_N^{\text{pr}}(\hat{\mathbf{u}}_N)) + \mathbf{z}_N^{\text{pr}\,T}\mathbf{J}_N^{\text{pr}}(\hat{\mathbf{u}}_N)(I - \tilde{\mathbf{J}}_N^{\text{pr}}(\hat{\mathbf{u}}_N)^{-1}\mathbf{J}_N^{\text{pr}}(\hat{\mathbf{u}}_N))\delta\hat{\mathbf{u}}_N + \mathcal{O}(\hat{\delta}^2) + \mathcal{O}(\tilde{\delta}^2)|\\ &= |-\mathbf{z}_N^{\text{pr}\,T}(I - (I - \mathbf{J}_N^{\text{pr}}(\hat{\mathbf{u}}_N)\tilde{\mathbf{J}}_N^{\text{pr}}(\hat{\mathbf{u}}_N)^{-1}))(\mathbf{r}_N^{\text{pr}}(\hat{\mathbf{u}}_N) - \tilde{\mathbf{r}}_N^{\text{pr}}(\hat{\mathbf{u}}_N))\\ &\qquad + \mathbf{z}_N^{\text{pr}\,T}(I - \mathbf{J}_N^{\text{pr}}(\hat{\mathbf{u}}_N)\tilde{\mathbf{J}}_N^{\text{pr}}(\hat{\mathbf{u}}_N)^{-1})\mathbf{J}_N^{\text{pr}}(\hat{\mathbf{u}}_N)\delta\hat{\mathbf{u}}_N + \mathcal{O}(\hat{\delta}^2) + \mathcal{O}(\tilde{\delta}^2)|\\ &\le |\mathbf{z}_N^{\text{pr}\,T}(\mathbf{r}_N^{\text{pr}}(\hat{\mathbf{u}}_N) - \tilde{\mathbf{r}}_N^{\text{pr}}(\hat{\mathbf{u}}_N))| + |\mathbf{z}_N^{\text{pr}\,T}\mathbf{A}(\mathbf{r}_N^{\text{pr}}(\hat{\mathbf{u}}_N) - \tilde{\mathbf{r}}_N^{\text{pr}}(\hat{\mathbf{u}}_N))| + |\mathbf{z}_N^{\text{pr}\,T}\mathbf{A}\mathbf{J}_N^{\text{pr}}(\hat{\mathbf{u}}_N)\delta\hat{\mathbf{u}}_N| + \mathcal{O}(\hat{\delta}^2) + \mathcal{O}(\tilde{\delta}^2).\end{aligned} \tag{17}$$

for $\mathbf{A} \equiv I - \mathbf{J}_N^{\text{pr}}(\hat{\mathbf{u}}_N)\tilde{\mathbf{J}}_N^{\text{pr}}(\hat{\mathbf{u}}_N)^{-1}$. The first term of (17) is bounded by

$$|\mathbf{z}_N^{\text{pr}\,T}(\mathbf{r}_N^{\text{pr}}(\hat{\mathbf{u}}_N) - \tilde{\mathbf{r}}_N^{\text{pr}}(\hat{\mathbf{u}}_N))| \le N\|\mathbf{z}_N^{\text{pr}} \circ (\mathbf{r}_N^{\text{pr}}(\hat{\mathbf{u}}_N) - \tilde{\mathbf{r}}_N^{\text{pr}}(\hat{\mathbf{u}}_N))\|_\infty \le \frac{2}{3}\delta^r, \tag{18}$$

where the first equality follows from $|\mathbf{a}^T\mathbf{b}| \le N\|\mathbf{a}\circ\mathbf{b}\|_\infty$ for all $\mathbf{a}, \mathbf{b} \in \mathbb{R}^N$, and the second inequality follows from $\mathbf{z}_N^{\text{pr}} \le \mathbf{z}_N^{\text{pr,w}}$ and (13). To bound the second term of (17), we first note that

$$z^{\text{pr,min}}\|\mathbf{r}_N^{\text{pr}}(\hat{\mathbf{u}}) - \tilde{\mathbf{r}}_N^{\text{pr}}(\hat{\mathbf{u}})\|_\infty \le \|\mathbf{z}_N^{\text{pr,w}} \circ (\mathbf{r}_N^{\text{pr}}(\hat{\mathbf{u}}) - \tilde{\mathbf{r}}_N^{\text{pr}}(\hat{\mathbf{u}}))\|_\infty \le \frac{2\delta^r}{3N}$$

by the definition of $\mathbf{z}_N^{\text{pr,w}}$, and hence

$$\|\mathbf{r}_N^{\text{pr}}(\hat{\mathbf{u}}) - \tilde{\mathbf{r}}_N^{\text{pr}}(\hat{\mathbf{u}})\|_\infty \le \frac{2\delta^r}{3Nz^{\text{pr,min}}} = \frac{2\sqrt{\delta^r}}{3N^{3/2}\|\mathbf{z}_N^{\text{pr}}\|_2} \tag{19}$$

by the definition of $z^{\text{pr,min}}$. It follows that the second term of (17) is bounded by

$$\begin{aligned}|\mathbf{z}_N^{\text{pr}\,T}\mathbf{A}(\mathbf{r}_N^{\text{pr}}(\hat{\mathbf{u}}_N) - \tilde{\mathbf{r}}_N^{\text{pr}}(\hat{\mathbf{u}}_N))| &= \|\mathbf{z}_N^{\text{pr}}\|_2\|\mathbf{A}\|_2\|\mathbf{r}_N^{\text{pr}}(\hat{\mathbf{u}}_N) - \tilde{\mathbf{r}}_N^{\text{pr}}(\hat{\mathbf{u}}_N)\|_2\\ &\le N^{3/2}\|\mathbf{z}_N^{\text{pr}}\|_2\|\mathbf{A}\|_{\max}\|\mathbf{r}_N^{\text{pr}}(\hat{\mathbf{u}}_N) - \tilde{\mathbf{r}}_N^{\text{pr}}(\hat{\mathbf{u}}_N)\|_\infty \le \frac{2}{3}\delta_{\mathbf{J}}\sqrt{\delta^r} \le \frac{1}{3}(\delta^r + \delta_{\mathbf{J}}^2),\end{aligned} \tag{20}$$

where the second to last inequality follows from (19) and the constraint (14), and the last inequality follows from Young's inequality. The third term of (17) is bounded by

$$|\mathbf{z}_N^{\mathrm{pr}\,T}\mathbf{A}\mathbf{J}_N^{\mathrm{pr}}(\hat{\mathbf{u}}_N)\delta\hat{\mathbf{u}}_N| \leq \|\mathbf{z}_N^{\mathrm{pr}}\|_2\|\mathbf{A}\|_2\|\mathbf{J}_N^{\mathrm{pr}}(\hat{\mathbf{u}}_N)\delta\hat{\mathbf{u}}_N\|_2 = \mathcal{O}(\delta_{\mathbf{J}}\hat{\delta}) = \mathcal{O}(\delta_{\mathbf{J}}^2) + \mathcal{O}(\hat{\delta}^2). \tag{21}$$

The substitution of (18), (20), and (21) into (17) yields

$$|\mathbf{q}_N(\mathbf{u}_N) - \mathbf{q}_N(\tilde{\mathbf{u}}_N)| \leq \delta^r + \mathcal{O}(\hat{\delta}^2) + \mathcal{O}(\tilde{\delta}^2) + \mathcal{O}(\delta_{\mathbf{J}}^2),$$

which is the desired inequality. □

We now relate the EQP residual constraint (11) to Proposition 1. We first note that the residual constraint (13) in Proposition 1 is precisely the residual manifold accuracy constraint (11) enforced for $\mu \in \Xi_J^{\mathrm{train}}$ in $\mathtt{LP}^r$. We second observe that the Jacobian constraint (14) is not directly enforced in our $\mathtt{LP}^r$. This Jacobian constraint measures the proximity of RB-EQP Jacobian $\tilde{\mathbf{J}}_N(\hat{\mathbf{u}}_N(\mu);\mu)$ to the RB counterpart $\mathbf{J}_N(\hat{\mathbf{u}}_N(\mu);\mu)$; while the constraint is not LP admissible as it requires the inverse of $\tilde{\mathbf{J}}_N(\hat{\mathbf{u}}_N(\mu);\mu)$, a closely related constraint $\|I - \mathbf{J}_N(\hat{\mathbf{u}}_N(\mu);\mu)^{-1}\tilde{\mathbf{J}}_N(\hat{\mathbf{u}}_N(\mu);\mu)\|_{\max}$ is in fact LP admissible. (The two constraints can be formally related using small-perturbation matrix inequalities.) Nevertheless, we do not enforce this Jacobian constraint in our EQP formulation as (a) the inclusion of the Jacobian constraint would increase the number of manifold accuracy constraints per training parameter point from N to $N + N^2$, (b) we hope the direct control of the EQP residual error by the constraint (11) also indirectly controls the related Jacobian error, and (c) in any event the effect of the Jacobian error on the output is second order (i.e., $\mathcal{O}(\delta_{\mathbf{J}}^2)$). In practice, as will be shown in a numerical example in Section 5, we have found that the EQP residual constraint (11) is sufficient to control the error $|\mathbf{q}_N(\mathbf{u}_N(\mu);\mu) - \mathbf{q}_N(\tilde{\mathbf{u}}_N(\mu);\mu)|$. (The Jacobian constraint was also omitted in[11] in the context of the $\mathcal{V}$-norm solution error control, instead of the output error control.)

Remark 4. In this work we consider two separate LPs: (i) $\mathtt{LP}^r$ with the constraints (11) for the residual, which yields $\{\rho_\kappa^r\}_{\kappa\in\mathcal{T}_h}$; (ii) $\mathtt{LP}^q$ with the constraints (12) for the output functional, which yields $\{\rho_\kappa^q\}_{\kappa\in\mathcal{T}_h}$. Alternatively, we could consider a single LP, $\mathtt{LP}^{rq}$, with constraints (11) and (12) and yield a single set of weights $\{\rho^{rq}\}_{\kappa\in\mathcal{T}_h}$ for both residual and output evaluation. In general, the number of nonzero (nnz) elements in these sets are related by $\mathrm{nnz}(\{\rho^r\}_{\kappa\in\mathcal{T}_h}) \leq \mathrm{nnz}(\{\rho^{rq}\}_{\kappa\in\mathcal{T}_h}) \leq \mathrm{nnz}(\{\rho_\kappa^r\}_{\kappa\in\mathcal{T}_h}) + \mathrm{nnz}(\{\rho_\kappa^q\}_{\kappa\in\mathcal{T}_h})$. The relationship indicates that the alternative formulation (a) reduces the memory footprint of the reduced model but (b) increases the number of elements involved in online residual evaluation, which increases the online cost. In this work, we use the two separate LPs to reduce the online cost.

3 | OUTPUT ERROR ESTIMATION

As discussed in the introduction, our goal in output error estimation is twofold. First, in the offline stage, we wish to estimate the output error in the FE snapshots, $|s(\mu) - s_h(\mu)|$, for all $\mu \in \Xi_N^{\mathrm{rb}}$. Second, in the online stage, we wish to estimate the error $|s_h(\mu) - \tilde{s}_N(\mu)|$ for any $\mu \in \mathcal{D}$; we in particular wish to compute the error estimate (i) in an online-efficient manner and (ii) for parameter values that in general do not belong to the training set Ξ_J^{train}. We address the offline FE error estimation procedure in Section 3.1 and the online RB(-EQP) error estimation procedure in Sections 3.2–3.4.

3.1 | FE output error estimation

We now wish to estimate the error in the FE output relative to the exact PDE output, $|s(\mu) - s_h(\mu)|$. To this end, we employ the DWR method[12]. We first introduce an enriched space for the surrogate adjoint $\mathcal{V}_{\hat{h}} \supset \mathcal{V}_h$; in this work we enrich the space by globally increasing the polynomial degree by one, i.e., $\mathcal{V}_{\hat{h}} \equiv \{v \in L^2(\Omega)^m \mid v \in \mathbb{P}^{p+1}(\kappa)^m,\ \forall\kappa \in \mathcal{T}_h\}$. We then introduce the dual problem: given $\mu \in \mathcal{D}$ and $u_h(\mu) \in \mathcal{V}_h$, find $z_{\hat{h}} \in \mathcal{V}_{\hat{h}}$ such that

$$r'_h(u_h(\mu); w, z_{\tilde{h}}(\mu); \mu) = q'_h(u_h(\mu); w; \mu) \quad \forall w \in \mathcal{V}_{\hat{h}},$$

where $r'_h(u_h(\mu); w, v; \mu)$ and $q'_h(u_h(\mu); w; \mu)$ are the Gâteaux derivatives of $r_h(\cdot, v; \mu)$ and $q_h(\cdot; \mu)$, respectively, about $u_h(\mu)$ in the direction w. The error in the output is then estimate by

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

The error in the error estimate arises from (i) the use of $u_h(\mu)$ as the linearization point of the adjoint problem (instead of the mean-value linearization between $u_h(\mu)$ and $u(\mu)$) and (ii) the approximation of the adjoint in $\mathcal{V}_{\tilde{h}}$ (instead of in the infinite-dimensional $\mathcal{V}$)[12]. In practice, the estimate has been used successfully in many aerodynamics applications; see a review paper[13].

3.2 | RB output error estimation

We now wish to estimate the error in the RB-EQP output $|s_h(\mu) - \tilde{s}_N(\mu)|$ for any $\mu \in \mathcal{D}$ in an online-efficient manner. To this end, we first invoke the triangle inequality to obtain

$$\begin{aligned} |s_h(\mu) - \tilde{s}_N(\mu)| &= |q_h(u_h(\mu);\mu) - \tilde{q}_h(\tilde{u}_N(\mu);\mu)| \\ &\le |q_h(u_h(\mu);\mu) - q_h(\tilde{u}_N(\mu);\mu)| + |q_h(\tilde{u}_N(\mu);\mu) - \tilde{q}_h(\tilde{u}_N(\mu);\mu)|; \end{aligned} \tag{22}$$

here, the first term is associated with the approximation of the state $u_h(\mu)$ by $\tilde{u}_N(\mu)$, and the second term is associated with the approximation of the functional output $q_h(\cdot;\cdot)$ by $\tilde{q}_h(\cdot;\cdot)$. As discussed in Section 2.7 and as we will also numerically demonstrate in Section 5, the second term is controlled using a tight EQP tolerance δ^q for the output functional evaluation. Hence, our goal in the online-efficient error estimation is to approximate $|q_h(u_h(\mu);\mu) - q_h(\tilde{u}_N(\mu);\mu)|$, the error in the output due to the use of the RB-EQP state $\tilde{u}_N(\mu)$ instead of the FE state $u_h(\mu)$.

To estimate $|q_h(u_h(\mu);\mu) - q_h(\tilde{u}_N(\mu);\mu)|$, we again employ the DWR method. However, as we wish the DWR error estimate to be online-efficient — i.e., computable in $\mathcal{O}(N)$ operations — the space in which the adjoint is approximated must be $\mathcal{O}(N)$-dimensional. Our approach is to approximate the adjoint by a linear combination of adjoint snapshots; this is an approach commonly used in existing goal-oriented RB formulations (for linear PDEs)[3]. Specifically, we introduce dual RB spaces $\mathcal{V}_N^{\mathrm{du}} \equiv \mathrm{span}\{z_h(\mu)\}_{\mu\in\Xi_N^{\mathrm{rb}}}$ associated with the RB parameter set Ξ_N^{rb} for $N = 1,\dots,N_{\max}$. Although in principle the primal and dual RB spaces may be associated with different RB parameter sets, in practice we use the same RB parameter set to reduce the offline computational cost. We then introduce an RB dual problem *linearized about the RB-EQP state*: given $\mu \in \mathcal{D}$ and $\tilde{u}_N(\mu) \in \mathcal{V}_N^{\mathrm{pr}}$, find $z_N^{\mathrm{du}}(\mu) \in \mathcal{V}_N^{\mathrm{du}}$ such that

$$r'(\tilde{u}_N(\mu); v, z_N^{\mathrm{du}}(\mu);\mu) = q'(\tilde{u}_N(\mu); v;\mu) \quad \forall v \in \mathcal{V}_N^{\mathrm{du}}. \tag{23}$$

The dual problem is linearized about the RB-EQP state $\tilde{u}_N(\mu) \in \mathcal{V}_N^{\mathrm{pr}}$ since we wish to estimate the error in $q_h(\tilde{u}_N(\mu);\mu)$ (and not $q_h(u_N(\mu);\mu)$). The output error of interest is then estimated by

$$|q_h(u_h(\mu);\mu) - q_h(\tilde{u}_N(\mu);\mu))| \approx \eta_N^{\mathrm{rb}}(\mu) \equiv |r(\tilde{u}_N(\mu), z_N^{\mathrm{du}}(\mu);\mu)| \tag{24}$$

We *assume* that the RB dual problem (23) is well posed and $z_N^{\mathrm{du}}(\mu) \notin \mathcal{V}_N^{\mathrm{pr}}$ so that the error estimate evaluates to a nonzero value.

Remark 5. Similarly to the FE error estimate discussed in Section 3.1, the difference in this error estimate and the true error $|q_h(u_h(\mu);\mu) - q_h(\tilde{u}_N(\mu);\mu))|$ arises from (i) the use of $\tilde{u}_N(\mu)$ as the linearization point of the adjoint problem (instead of the mean-value linearization between $\tilde{u}_N(\mu)$ and $u_h(\mu)$) and (ii) the approximation of the adjoint in $\mathcal{V}_N^{\mathrm{du}}$ (instead of in $\mathcal{V}_h$). In particular, the error due to (ii) could be significant for certain problems; this error estimate in fact does not work when the primal and dual solutions are identical so that $\mathcal{V}_N^{\mathrm{pr}} = \mathrm{span}\{u_h(\mu)\}_{\mu\in\Xi_N^{\mathrm{rb}}} = \mathrm{span}\{z_h(\mu)\}_{\mu\in\Xi_N^{\mathrm{rb}}} = \mathcal{V}_N^{\mathrm{du}}$ (e.g., the compliance output for linear elasticity, which is self-adjoint), as the error estimate would evaluate to zero by Galerkin orthogonality. In principle, we could introduce an enriched RB dual space $\mathcal{V}_{N^{\mathrm{du}}}^{\mathrm{du}}$ for $N^{\mathrm{du}} > N$, in a manner similar to the FE counterpart with an enriched p, to make the error estimate more reliable. However, for typical output prediction problems in aerodynamics, we have found the error estimate associated with $\mathcal{V}_N^{\mathrm{du}}$ of dimension N to work well, as we demonstrate in Section 5.

We now recast the problem in a discrete form. To this end, we first introduce a $\mathcal{V}$-orthonormal basis $\{\phi_i^{\mathrm{du}}\}_{i=1}^N$ of $\mathcal{V}_N^{\mathrm{du}}$. We next introduce an operator $V_N^{\mathrm{du}} : \mathbb{R}^N \to \mathcal{V}_N^{\mathrm{du}}$ which associates a generalized coordinate $\mathbf{v} \in \mathbb{R}^N$ to $v = V_N^{\mathrm{du}}\mathbf{v} = \sum_{j=1}^N \mathbf{v}_j \phi_j^{\mathrm{du}} \in \mathcal{V}_N^{\mathrm{du}}$. We then introduce a Jacobian and output gradient evaluated with respect to the dual RB: $\mathbf{J}_N^{\mathrm{du}} : \mathbb{R}^N \times \mathcal{D} \to \mathbb{R}^{N\times N}$ and $\mathbf{g}_N^{\mathrm{du}} : \mathbb{R}^N \times \mathcal{D} \to \mathbb{R}^N$ such that

$$\begin{aligned} \mathbf{J}_N^{\mathrm{du}}(\mathbf{w};\mu)_{ij} &\equiv r_h'(V_N^{\mathrm{pr}}\mathbf{w}; \phi_j^{\mathrm{du}}, \phi_i^{\mathrm{du}};\mu), \quad i,j = 1,\dots,N, \\ \mathbf{g}_N^{\mathrm{du}}(\mathbf{w};\mu)_j &\equiv q_h'(V_N^{\mathrm{pr}}\mathbf{w}; \phi_j^{\mathrm{du}};\mu), \quad j = 1,\dots,N. \end{aligned}$$

We note that these operators are *not* the discrete Jacobian and gradient:

$$\mathbf{J}_N^{\mathrm{du}}(\mathbf{u}_N(\mu);\mu)_{ij} \neq \left.\frac{\partial \mathbf{r}_{N,i}^{\mathrm{pr}}}{\partial \mathbf{w}_j}\right|_{(\mathbf{u}_N(\mu);\mu)} \quad \text{and} \quad \mathbf{g}_N^{\mathrm{du}}(\mathbf{u}_N(\mu);\mu)_j \neq \left.\frac{\partial \mathbf{q}_N}{\partial \mathbf{w}_j}\right|_{(\mathbf{u}_N(\mu);\mu)}.$$

The linearization points of $\mathbf{J}^{\rm du}_N(\mathbf{w};\mu)$ and $\mathbf{g}^{\rm du}_N(\mathbf{w};\mu)$ are in $\mathcal{V}^{\rm pr}_N$ but the Gâteaux derives are evaluted in $\mathcal{V}^{\rm du}_N$; in other words, this is an RB approximation of the continuous dual problem and not the dual of the discrete RB problem. The RB dual problem is as follows: given $\mu \in \mathcal{D}$ and $\tilde{\mathbf{u}}_N(\mu) \in \mathbb{R}^N$, find $\mathbf{z}^{\rm du}_N(\mu) \in \mathbb{R}^N$ such that

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

note that the problem is again linearized about $\tilde{u}_N(\mu)$ (and not $u_N(\mu)$), and $z^{\rm du}_N(\mu) = V^{\rm du}_N \mathbf{z}^{\rm du}_N(\mu)$. To provide a discrete expression for the output error of interest, we first introduce the discrete RB residual evaluated with respect to the dual RB: $\mathbf{r}^{\rm du} : \mathbb{R}^N \times \mathcal{D} \to \mathbb{R}^N$ such that

$$\mathbf{r}^{\rm du}_N(\mathbf{w};\mu)_i \equiv r_h(V^{\rm pr}_N \mathbf{w}, \phi^{\rm du}_i;\mu), \quad i = 1,\dots,N.$$

The error estimate is given by

$$\eta^{\rm rb}_N(\mu) \equiv |\mathbf{z}^{\rm du}_N(\mu)^T \mathbf{r}^{\rm du}_N(\tilde{\mathbf{u}}_N(\mu);\mu)|,$$

which is simply a discrete form of (24).

Remark 6. Because the DG approximation is dual consistent, the solution $z^{\rm pr}_N \in \mathcal{V}^{\rm pr}_N$ to the dual of the discrete RB problem associated with $\mathcal{V}^{\rm pr}_N$ also provides a consistent approximation of the continuous dual problem. However, $z^{\rm pr}_N \in \mathcal{V}^{\rm pr}_N$ may converge slowly to z, or may not converge at all, as the space $\mathcal{V}^{\rm pr}_N$ is tailored for the primal solution and not the dual solution. In addition, $\eta^{\rm rb}_N$ would evaluate to zero if $z^{\rm pr}_N$ is used due to Galerkin orthogonality. Hence we employ $z^{\rm du}_N \in \mathcal{V}^{\rm du}_N$ which converges rapidly to z (or more precisely z_h) and provides an effective error estimate.

3.3 | RB-EQP output error estimation

The error estimate $\eta^{\rm rb}_N(\mu)$ is not online-efficient because (a) the RB dual problem (23) requires the evaluation of the FE Jacobian and output forms and (b) the error estimate (24) requires the evaluation of the FE residual form. We now consider EQP approximations of these problems. To this end, we introduce EQP approximations of the residual, Jacobian, and output gradient forms associated with a given set of EQP weights $\{\rho^\eta_\kappa \in \mathbb{R}_{\geq 0}\}_{\kappa\in\mathcal{T}_h}$:

$$\begin{aligned}
\tilde{r}_h(w,v;\mu) &\equiv \sum_{\kappa\in\mathcal{T}_h} \rho^\eta_\kappa r_\kappa(w,v;\mu),\\
\tilde{r}'_h(y;w,v;\mu) &\equiv \sum_{\kappa\in\mathcal{T}_h} \rho^\eta_\kappa r'_\kappa(y;w,v;\mu),\\
\tilde{q}'_h(y;w;\mu) &\equiv \sum_{\kappa\in\mathcal{T}_h} \rho^\eta_\kappa q'_\kappa(y;w;\mu);
\end{aligned}$$

the superscript η on ρ^η_κ signifies the EQP weight is associated with the error estimate $\eta^{\rm rb}_N(\mu)$. We then introduce an RB-EQP dual problem: given $\mu \in \mathcal{D}$ and $\tilde{u}_N(\mu) \in \mathcal{V}^{\rm pr}_N$, find $\tilde{z}^{\rm du}_N(\mu) \in \mathcal{V}^{\rm du}_N$ such that

$$\tilde{r}'_h(\tilde{u}_N(\mu); v, \tilde{z}^{\rm du}_N(\mu);\mu) = \tilde{q}'_h(\tilde{u}_N(\mu); v;\mu) \quad \forall v \in \mathcal{V}^{\rm du}_N. \tag{25}$$

Our output error estimate is given by

$$|q_h(u_h(\mu);\mu) - q_h(\tilde{u}_N(\mu);\mu)| \approx \tilde{\eta}^{\rm rb}_N(\mu) \equiv |\tilde{r}_h(\tilde{u}_N(\mu), \tilde{z}^{\rm du}_N(\mu);\mu)|. \tag{26}$$

We again *assume* $\tilde{z}^{\rm du}_N(\mu) \notin \mathcal{V}^{\rm pr}_N$ such that the error estimate evaluates to a nonzero value; see Remark 5.

We now recast the problem in a discrete form. To introduce the dual problem, we first introduce EQP approximations of the Jacobian and output gradient evaluated with respect to the dual RB: $\tilde{\mathbf{J}}^{\rm du}_N : \mathbb{R}^N \times \mathcal{D} \to \mathbb{R}^{N\times N}$ and $\tilde{\mathbf{g}}^{\rm du}_N : \mathbb{R}^N \times \mathcal{D} \to \mathbb{R}^N$ such that

$$\begin{aligned}
\tilde{\mathbf{J}}^{\rm du}_N(\mathbf{w};\mu)_{ij} &\equiv \tilde{r}'_h(V^{\rm pr}_N \mathbf{w}; \phi^{\rm du}_j, \phi^{\rm du}_i;\mu), \quad i,j = 1,\dots,N,\\
\tilde{\mathbf{g}}^{\rm du}_N(\mathbf{w};\mu)_j &\equiv \tilde{q}'_h(V^{\rm pr}_N \mathbf{w}; \phi^{\rm du}_j;\mu), \quad j = 1,\dots,N.
\end{aligned}$$

The RB-EQP dual problem is as follows: given $\mu \in \mathcal{D}$ and $\tilde{\mathbf{u}}_N(\mu) \in \mathbb{R}^N$, find $\tilde{\mathbf{z}}^{\rm du}_N(\mu) \in \mathbb{R}^N$ such that

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

note that $\tilde{z}^{\rm du}_N(\mu) = V^{\rm du}_N \tilde{\mathbf{z}}^{\rm du}_N(\mu)$. To provide a discrete expression for the output error estimate $\tilde{\eta}^{\rm rb}_N(\mu)$, we then introduce the discrete RB-EQP residual evaluated with respect to the dual RB: $\tilde{\mathbf{r}}^{\rm du}_N : \mathbb{R}^N \times \mathcal{D} \to \mathbb{R}^N$ such that

$$\tilde{\mathbf{r}}^{\rm du}_N(\mathbf{w};\mu)_i \equiv \tilde{r}_h(V^{\rm pr}_N \mathbf{w}, \phi^{\rm du}_i;\mu), \quad i = 1,\dots,N.$$

The output error estimate is given by

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

which is a discrete form of (26).

Following the goal set forth in Section 3.2, the online-efficient output error estimate $\tilde{\eta}_N^{\mathrm{rb}}(\mu)$ given by (26) approximates the error in the output due to the RB-EQP approximation of the state $|q_h(u_h(\mu);\mu) - q_h(\tilde{u}_N(\mu);\mu)|$. The error estimate $\tilde{\eta}_N^{\mathrm{rb}}(\mu)$ does not account for the output error due to the EQP approximation of the output functional $|q_h(\tilde{u}_N(\mu);\mu) - \tilde{q}_h(\tilde{u}_N(\mu);\mu)|$, the other component of the overall output error in (22); however, this error is controlled by using a tight output functional EQP tolerance δ^q in the offline stage, as discussed further in Section 4.2.

3.4 | EQP: DWR accuracy constraints

We now wish to identify EQP constraints such that the error in the error estimates due to hyperreduction, $|\eta_N^{\mathrm{rb}}(\mu) - \tilde{\eta}_N^{\mathrm{rb}}(\mu)|$, is controlled to the user-specified tolerance $\delta^\eta \in \mathbb{R}_{>0}$. To this end, we first note a decomposition of the hyperreduction error:

$$\begin{aligned}\eta_N^{\mathrm{rb}}(\mu) - \tilde{\eta}_N^{\mathrm{rb}}(\mu) &= \mathbf{z}_N^{\mathrm{du}}(\mu)^T \mathbf{r}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N(\mu);\mu) - \tilde{\mathbf{z}}_N^{\mathrm{du}}(\mu)^T \tilde{\mathbf{r}}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N(\mu);\mu)\\ &= \mathbf{z}_N^{\mathrm{du}}(\mu)^T (\mathbf{r}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N(\mu);\mu) - \tilde{\mathbf{r}}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N(\mu);\mu)) + (\mathbf{z}_N^{\mathrm{du}}(\mu) - \tilde{\mathbf{z}}_N^{\mathrm{du}}(\mu))^T \mathbf{r}_N^{\mathrm{du}}(\tilde{u}_N(\mu);\mu)\\ &\quad + (\mathbf{z}_N^{\mathrm{du}}(\mu) - \tilde{\mathbf{z}}_N^{\mathrm{du}}(\mu))^T (\mathbf{r}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N(\mu);\mu) - \tilde{\mathbf{r}}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N(\mu);\mu)).\end{aligned}$$

The decomposition shows that the hyperreduction error in the error estimate comprises three terms: (i) the error in the residual evaluation; (ii) the error in the adjoint approximation; and (iii) the product of the two errors. Our plan is to introduce two sets of EQP constraints whose primary purpose is to control (i) and (ii), which are first order; the same set of constraints, with little modifications, will also control (iii), which is second order. Specifically, we wish to ensure that sources (i) and (ii) are each less than $\delta^\eta/2$ so that their sum is less than the user-specific tolerance δ^η.

We first introduce scaling constants required for the modification to control the second-order error:

$$\alpha(\mu) \equiv \frac{\|\mathbf{z}_N^{\mathrm{du}}(\mu)\|_2}{\|\mathbf{r}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N(\mu))\|_2}, \quad z^{\mathrm{du,min}}(\mu) \equiv \frac{1}{2}\sqrt{\frac{\alpha(\mu)\delta^\eta}{N}}, \quad \text{and} \quad r^{\mathrm{du,min}}(\mu) \equiv \frac{1}{2}\sqrt{\frac{\delta^\eta}{\alpha(\mu)N}}.$$

We then define

$$\begin{aligned}\mathbf{z}_N^{\mathrm{du,w}}(\mu) &= \max\{|\mathbf{z}_N^{\mathrm{du}}(\mu)|, z^{\mathrm{du,min}}(\mu)\},\\ \mathbf{r}_N^{\mathrm{du,w}}(\tilde{\mathbf{u}}_N(\mu);\mu) &= \max\{|\mathbf{r}_N^{\mathrm{du}}(\tilde{u}_N(\mu);\mu)|, r^{\mathrm{du,min}}(\mu)\},\end{aligned}$$

where the max operator between the vector and scalar is applied to each element of the vector. These are the modified dual and residual "weights" with minimum value constraints and are denoted by the superscript "w".

We now introduce the constraints. To control the error due to source (i) — the approximation of $\mathbf{r}_N^{\mathrm{du}}(\cdot;\mu)$ —, we introduce an (primal) residual EQP constraint: for $i = 1, \dots, N$ and for all $\mu \in \Xi_J^{\mathrm{train}}$,

$$|\mathbf{z}_N^{\mathrm{du,w}}(\mu) \circ (\mathbf{r}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N(\mu);\mu) - \sum_{\kappa \in \mathcal{T}_h} \rho_\kappa^\eta \mathbf{r}_{N,\kappa}^{\mathrm{du}}(\tilde{\mathbf{u}}_N(\mu);\mu))|_i \le \frac{\delta^\eta}{2N}. \tag{27}$$

To control the error due to source (ii) — the approximation of the dual solution $\tilde{\mathbf{z}}_N^{\mathrm{du}}(\mu)$ —, we impose a set of two constraints: for $i = 1, \dots, N$ and for all $\mu \in \Xi_J^{\mathrm{train}}$,

$$|\mathbf{r}_N^{\mathrm{du,w}}(\tilde{\mathbf{u}}_N(\mu);\mu) \circ \mathbf{J}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N(\mu);\mu)^{-T} (\mathbf{J}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N(\mu);\mu)^T \mathbf{z}^{\mathrm{du}}(\mu) - \sum_{\kappa \in \mathcal{T}_h} \rho_\kappa^\eta \mathbf{J}_{N,\kappa}^{\mathrm{du}}(\tilde{\mathbf{u}}_N(\mu);\mu)^T \mathbf{z}^{\mathrm{du}}(\mu))|_i \le \frac{\delta^\eta}{4N}, \tag{28}$$

$$|\mathbf{r}_N^{\mathrm{du,w}}(\tilde{\mathbf{u}}_N(\mu);\mu) \circ \mathbf{J}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N(\mu);\mu)^{-T} (\mathbf{g}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N(\mu);\mu) - \sum_{\kappa \in \mathcal{T}_h} \rho_\kappa^\eta \mathbf{g}_{N,\kappa}(\tilde{\mathbf{u}}_N^{\mathrm{du}}(\mu);\mu))|_i \le \frac{\delta^\eta}{4N}. \tag{29}$$

These sets of constraints, thanks to the replacement of $\mathbf{z}_N^{\mathrm{du}}(\mu)$ and $\mathbf{r}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N(\mu);\mu)$ with $\mathbf{z}_N^{\mathrm{du,w}}(\mu)$ and $\mathbf{r}_N^{\mathrm{du,w}}(\tilde{\mathbf{u}}_N(\mu);\mu)$, will also control the error due to source (iii), the second-order term. The constraints (27)–(29) can be written as linear inequality constraints in $\{\rho_\kappa^\eta\}_{\kappa \in \mathcal{T}_h}$ using Remark 2 and hence are admissible in LP. In addition, the LP has a feasible solution for any $\Xi_J^{\mathrm{train}} \subset \mathcal{D}$, as the left hand side of the constraints evaluates to 0 for the exact (non-sparse) weights $\{\rho_\kappa^\eta = 1\}_{\kappa \in \mathcal{T}_h}$. The LP procedure for the DWR, denoted by $\mathtt{LP}^\eta(\Xi_J^{\mathrm{train}}, U_J^{\mathrm{train}} \equiv \{\tilde{\mathbf{u}}_N(\mu)\}_{\mu \in \Xi_J^{\mathrm{train}}}, \delta^\eta)$, is obtained by the substitution of the three DWR manifold accuracy constrains (27)–(29) in the place of the generic manifold accuracy constraint (10) of the generic LP procedure $\mathtt{LP}^\bullet$. The number of manifold accuracy constraints per training parameter point is $M = 3N$; the total number of manifold accuracy constraints for the J training parameter points is $3NJ$ (which can be written as $6NJ$ linear inequality constraints following Remark 2).

We now introduce a proposition that relates the constraint (27) to the error due to source (i).

Proposition 2 (DWR error due to approximation of $\mathbf{r}_N^{\mathrm{du}}(\cdot;\cdot)$)**.** Suppose

$$\|\mathbf{z}_N^{\mathrm{du,w}} \circ (\mathbf{r}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N) - \tilde{\mathbf{r}}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N))\|_\infty \leq \frac{\delta^\eta}{2N}. \tag{30}$$

Then

$$|(\mathbf{z}_N^{\mathrm{du}})^T (\mathbf{r}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N) - \tilde{\mathbf{r}}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N))| \leq \frac{\delta^\eta}{2}, \tag{31}$$

$$\|\mathbf{r}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N) - \tilde{\mathbf{r}}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N)\|_2 \leq \sqrt{\frac{\delta^\eta}{\alpha}}. \tag{32}$$

Proof. The inequality (31) follows from

$$|(\mathbf{z}_N^{\mathrm{du}})^T (\mathbf{r}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N) - \tilde{\mathbf{r}}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N))| \leq N \|\mathbf{z}_N^{\mathrm{du,w}} \circ (\mathbf{r}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N) - \tilde{\mathbf{r}}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N))\|_\infty \leq \frac{\delta^\eta}{2},$$

where the first inequality follows from $|\mathbf{z}_N^{\mathrm{du}}|_i \leq |\mathbf{z}_N^{\mathrm{du,w}}|_i$ for $i = 1, \dots, N$ and the fact that $|\mathbf{a}^T\mathbf{b}| \leq N \|\mathbf{a} \circ \mathbf{b}\|_\infty$ for any $\mathbf{a}, \mathbf{b} \in \mathbb{R}^N$, and the second inequality follows from (30).

The inequality (32) follows from

$$\|z^{\min} (\mathbf{r}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N) - \tilde{\mathbf{r}}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N))\|_2 \leq \sqrt{N} \|\mathbf{z}_N^{\mathrm{du,w}} \circ (\mathbf{r}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N) - \tilde{\mathbf{r}}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N))\|_\infty \leq \frac{\delta^\eta}{2\sqrt{N}},$$

where the first inequality follows from $z^{\min} \leq |\mathbf{z}_N^{\mathrm{du,w}}|_i$ for $i = 1, \dots, N$ and $\|\mathbf{a}\|_2 \leq \sqrt{N} \|\mathbf{a}\|_\infty$ for any $\mathbf{a} \in \mathbb{R}^N$, and the second inequality follows from (30). The division of the equation by $z^{\min} \equiv \frac{1}{2}\sqrt{\alpha\delta^\eta/N}$ gives the desired result. □

We note that the constraint (30) in Proposition (2) is precisely the first DWR manifold accuracy constraint (27) enforced for $\mu \in \Xi_J^{\mathrm{train}}$ in $\mathtt{LP}^\eta$. Hence, the first DWR constraint (27) controls the error due to source (i) to $\delta^\eta/2$ for $\mu \in \Xi_J^{\mathrm{train}}$.

We next introduce a proposition that relates the constraints (28) and (29) to the error due to source (ii).

Proposition 3. Suppose

$$\|\mathbf{r}_N^{\mathrm{du,w}}(\tilde{\mathbf{u}}_N) \circ (\mathbf{J}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N)^{-T} (\mathbf{J}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N)^T \mathbf{z}_N^{\mathrm{du}} - \tilde{\mathbf{J}}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N)^T \mathbf{z}_N^{\mathrm{du}}))\|_\infty \leq \frac{\delta^\eta}{4N}, \tag{33}$$

$$\|\mathbf{r}_N^{\mathrm{du,w}}(\tilde{\mathbf{u}}_N) \circ (\mathbf{J}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N)^{-T} (\mathbf{g}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N) - \tilde{\mathbf{g}}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N)))\|_\infty \leq \frac{\delta^\eta}{4N}, \tag{34}$$

$$\|I - \tilde{\mathbf{J}}_N^{\mathrm{du}}(\hat{\mathbf{u}}_N)^{-T} \mathbf{J}_N^{\mathrm{du}}(\hat{\mathbf{u}}_N)^T\|_{\max} \leq \delta_{\mathbf{J}}, \tag{35}$$

where $\|A\|_{\max}$ denotes the maximum entry of the matrix $A \in \mathbb{R}^{N \times N}$. Then

$$|\mathbf{r}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N)^T (\mathbf{z}_N^{\mathrm{du}} - \tilde{\mathbf{z}}_N^{\mathrm{du}})| \leq \frac{\delta^\eta}{2} + \mathcal{O}((\delta^\eta)^2) + \mathcal{O}(\delta_{\mathbf{J}}^2), \tag{36}$$

$$\|\mathbf{z}_N^{\mathrm{du}} - \tilde{\mathbf{z}}_N^{\mathrm{du}}\|_2 \leq \frac{1}{2}\sqrt{\alpha\delta^\eta} + \mathcal{O}(\delta^\eta) + \mathcal{O}(\delta_{\mathbf{J}}^2). \tag{37}$$

Proof. For notational brevity, we omit $\tilde{\mathbf{u}}_N$ in the argument for $\mathbf{r}_N^{\mathrm{du}}$, $\mathbf{J}_N^{\mathrm{du}}$, and $\mathbf{g}_N^{\mathrm{du}}$ as these operators are always evaluated about $\tilde{\mathbf{u}}_N$. We first prove (36). We recall $\tilde{\mathbf{z}}_N^{\mathrm{du}} = \tilde{\mathbf{J}}_N^{\mathrm{du}-T} \tilde{\mathbf{g}}_N^{\mathrm{du}}$ and obtain

$$\begin{aligned}\mathbf{z}_N^{\mathrm{du}} - \tilde{\mathbf{z}}_N^{\mathrm{du}} &= \mathbf{z}_N^{\mathrm{du}} - \tilde{\mathbf{J}}_N^{\mathrm{du}-T} \tilde{\mathbf{g}}_N^{\mathrm{du}} = \tilde{\mathbf{J}}_N^{\mathrm{du}-T} (\tilde{\mathbf{J}}_N^{\mathrm{du}T} \mathbf{z}_N^{\mathrm{du}} - \tilde{\mathbf{g}}_N^{\mathrm{du}}) = (I - (I - \tilde{\mathbf{J}}_N^{\mathrm{du}-T} \mathbf{J}_N^{\mathrm{du}T})) \mathbf{J}_N^{\mathrm{du}-T} (\tilde{\mathbf{J}}_N^{\mathrm{du}T} \mathbf{z}_N^{\mathrm{du}} - \tilde{\mathbf{g}}_N^{\mathrm{du}}) \\ &= (I - \mathbf{A}) \mathbf{J}_N^{\mathrm{du}-T} (\tilde{\mathbf{J}}_N^{\mathrm{du}T} \mathbf{z}_N^{\mathrm{du}} - \tilde{\mathbf{g}}_N^{\mathrm{du}}),\end{aligned} \tag{38}$$

where $\mathbf{A} \equiv I - \tilde{\mathbf{J}}_N^{\mathrm{du}-T} \mathbf{J}_N^{\mathrm{du}T}$. It hence follows that

$$\begin{aligned}|\mathbf{r}_N^{\mathrm{du}T} (\mathbf{z}_N^{\mathrm{du}} - \tilde{\mathbf{z}}_N^{\mathrm{du}})| &= |\mathbf{r}_N^{\mathrm{du}T} (I - \mathbf{A}) \mathbf{J}_N^{\mathrm{du}-T} (\tilde{\mathbf{J}}_N^{\mathrm{du}T} \mathbf{z}_N^{\mathrm{du}} - \tilde{\mathbf{g}}_N^{\mathrm{du}})| \\ &\leq |\mathbf{r}_N^{\mathrm{du}T} \mathbf{J}_N^{\mathrm{du}-T} (\tilde{\mathbf{J}}_N^{\mathrm{du}T} \mathbf{z}_N^{\mathrm{du}} - \tilde{\mathbf{g}}_N^{\mathrm{du}})| + |\mathbf{r}_N^{\mathrm{du}T} \mathbf{A} \mathbf{J}_N^{\mathrm{du}-T} (\tilde{\mathbf{J}}_N^{\mathrm{du}T} \mathbf{z}_N^{\mathrm{du}} - \tilde{\mathbf{g}}_N^{\mathrm{du}})|.\end{aligned} \tag{39}$$

To bound the first term of (39), we combine (33) and (34) to obtain

$$\begin{aligned}\|\mathbf{r}_N^{\mathrm{du,w}} \circ (\mathbf{J}_N^{\mathrm{du}-T} (\tilde{\mathbf{J}}_N^{\mathrm{du}T} \mathbf{z}_N^{\mathrm{du}} - \tilde{\mathbf{g}}_N^{\mathrm{du}}))\|_\infty &= \|\mathbf{r}_N^{\mathrm{du,w}} \circ \mathbf{J}_N^{\mathrm{du}-T} ((\tilde{\mathbf{J}}_N^{\mathrm{du}T} \mathbf{z}_N^{\mathrm{du}} - \tilde{\mathbf{g}}_N^{\mathrm{du}}) - (\mathbf{J}_N^{\mathrm{du}T} \mathbf{z}_N^{\mathrm{du}} - \mathbf{g}_N^{\mathrm{du}}))\|_\infty \\ &= \|\mathbf{r}_N^{\mathrm{du,w}} \circ \mathbf{J}_N^{\mathrm{du}-T} ((\tilde{\mathbf{J}}_N^{\mathrm{du}T} \mathbf{z}_N^{\mathrm{du}} - \mathbf{J}_N^{\mathrm{du}T} \mathbf{z}_N^{\mathrm{du}}) - (\tilde{\mathbf{g}}_N^{\mathrm{du}} - \mathbf{g}_N^{\mathrm{du}}))\|_\infty \leq \frac{\delta^\eta}{2N},\end{aligned}$$

and hence

$$|\mathbf{r}_N^{\mathrm{du}T}\mathbf{J}_N^{\mathrm{du}-T}(\tilde{\mathbf{J}}_N^{\mathrm{du}T}\mathbf{z}_N^{\mathrm{du}} - \tilde{\mathbf{g}}_N^{\mathrm{du}})| \le N\|\mathbf{r}_N^{\mathrm{du}}\circ(\mathbf{J}_N^{\mathrm{du}-T}(\tilde{\mathbf{J}}_N^{\mathrm{du}T}\mathbf{z}_N^{\mathrm{du}} - \tilde{\mathbf{g}}_N^{\mathrm{du}}))\|_\infty \le \delta^\eta/2. \tag{40}$$

To bound the second term of (39), we note the following:

$$r^{\mathrm{du,min}}\|\mathbf{J}_N^{\mathrm{du}-T}(\tilde{\mathbf{J}}_N^{\mathrm{du}T}\mathbf{z}_N^{\mathrm{du}} - \tilde{\mathbf{g}}_N^{\mathrm{du}})\|_\infty \le \|\mathbf{r}_N^{\mathrm{du,w}}\circ\mathbf{J}_N^{\mathrm{du}-T}(\tilde{\mathbf{J}}_N^{\mathrm{du}T}\mathbf{z}_N^{\mathrm{du}} - \tilde{\mathbf{g}}_N^{\mathrm{du}})\|_\infty \le \frac{\delta^\eta}{4N}$$

by $r^{\mathrm{du,min}} \le \mathbf{r}_N^{\mathrm{du,w}}$ and (33); it follows that

$$\|\mathbf{J}_N^{\mathrm{du}-T}(\tilde{\mathbf{J}}_N^{\mathrm{du}T}\mathbf{z}_N^{\mathrm{du}} - \tilde{\mathbf{g}}_N^{\mathrm{du}})\|_\infty \le \frac{\delta^\eta}{4Nr^{\mathrm{du,min}}} \le \frac{\sqrt{\alpha\delta^\eta}}{2\sqrt{N}} = \frac{\sqrt{\delta^\eta\|\mathbf{z}_N^{\mathrm{du}}\|_2}}{2\sqrt{N\|\mathbf{r}_N^{\mathrm{du}}\|_2}} \tag{41}$$

by the definition of $r^{\mathrm{du,min}}$ and α; and hence

$$\begin{aligned}|\mathbf{r}_N^{\mathrm{du}T}\mathbf{A}\mathbf{J}_N^{\mathrm{du}-T}(\tilde{\mathbf{J}}_N^{\mathrm{du}T}\mathbf{z}_N^{\mathrm{du}} - \tilde{\mathbf{g}}_N^{\mathrm{du}})| &\le \|\mathbf{r}_N^{\mathrm{du}}\|_2\|\mathbf{A}\|_2\|\mathbf{J}_N^{\mathrm{du}-T}(\tilde{\mathbf{J}}_N^{\mathrm{du}T}\mathbf{z}_N^{\mathrm{du}} - \tilde{\mathbf{g}}_N^{\mathrm{du}})\|_2\\ &\le N^{3/2}\|\mathbf{r}_N^{\mathrm{du}}\|_2\|\mathbf{A}\|_{\max}\|\mathbf{J}_N^{\mathrm{du}-T}(\tilde{\mathbf{J}}_N^{\mathrm{du}T}\mathbf{z}_N^{\mathrm{du}} - \tilde{\mathbf{g}}_N^{\mathrm{du}})\|_\infty \le \frac{N}{2}\sqrt{\|\mathbf{r}_N^{\mathrm{du}}\|_2\|\mathbf{z}_N^{\mathrm{du}}\|_2}\sqrt{\delta^\eta}\delta_{\mathbf{J}} \le \epsilon\delta^\eta + \mathcal{O}(\delta_{\mathbf{J}}^2)\end{aligned} \tag{42}$$

for an arbitrary small ϵ by Young's inequality. The substitution of (40) and (42) into (39) yields

$$|\mathbf{r}_N^{\mathrm{du}T}(\mathbf{z}_N^{\mathrm{du}} - \tilde{\mathbf{z}}_N^{\mathrm{du}})| \le (\frac{1}{2} + \epsilon)\delta^\eta + \mathcal{O}(\delta_{\mathbf{J}}^2)$$

for ϵ arbitrarily small, which is the desired inequality (36).

We now prove the second inequality (37):

$$\begin{aligned}\|\mathbf{z}_N^{\mathrm{du}} - \tilde{\mathbf{z}}_N^{\mathrm{du}}\|_2 &= \|(I - \mathbf{A})\mathbf{J}_N^{\mathrm{du}-T}(\tilde{\mathbf{J}}_N^{\mathrm{du}T}\mathbf{z}_N^{\mathrm{du}} - \tilde{\mathbf{g}}_N^{\mathrm{du}})\|_2 \le (1 + \|\mathbf{A}\|_2)\|\mathbf{J}_N^{\mathrm{du}-T}(\tilde{\mathbf{J}}_N^{\mathrm{du}T}\mathbf{z}_N^{\mathrm{du}} - \tilde{\mathbf{g}}_N^{\mathrm{du}})\|_2\\ &\le (1 + N\|\mathbf{A}\|_{\max})\sqrt{N}\|\mathbf{J}_N^{\mathrm{du}-T}(\tilde{\mathbf{J}}_N^{\mathrm{du}T}\mathbf{z}_N^{\mathrm{du}} - \tilde{\mathbf{g}}_N^{\mathrm{du}})\|_\infty \le \frac{1}{2}(1 + N\delta_{\mathbf{J}})\sqrt{\alpha\delta^\eta} \le \frac{1}{2}\sqrt{\alpha\delta^\eta} + \mathcal{O}(\delta^\eta) + \mathcal{O}(\delta_{\mathbf{J}}^2);\end{aligned}$$

here, the first equality follows from (38), the first inequality follows from $\|(I - \mathbf{A})\mathbf{v}\|_2 \le (1 + \|\mathbf{A}\|_2)\|\mathbf{v}\|_2\ \forall\mathbf{v} \in \mathbb{R}^N$, the second inequality follows from the norm inequalities, the third inequality follow from (35) and (41), and the last inequality follows from Young's inequality. □

Corollary 1. The combination of (32) and (37) in Propositions 2 and 3, respectively, implies that

$$|(\mathbf{z}_N^{\mathrm{du}} - \tilde{\mathbf{z}}_N^{\mathrm{du}})^T(\mathbf{r}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N) - \tilde{\mathbf{r}}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N))| \le \|\mathbf{z}_N^{\mathrm{du}} - \tilde{\mathbf{z}}_N^{\mathrm{du}}\|_2\|\mathbf{r}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N) - \tilde{\mathbf{r}}_N^{\mathrm{du}}(\tilde{\mathbf{u}}_N)\|_2 \le \delta^\eta/2;$$

the higher-order error is at most $\delta^\eta/2$ (and is nominally $(\delta^\eta)^2$).

We now relate the EQP DWR constraints (28) and (29) to Proposition (3). We first note that the two constraints (33) and (34) in Proposition (3) are precisely the DWR manifold accuracy constraints (28) and (29) enforced for $\mu \in \Xi_J^{\mathrm{train}}$ in $\mathtt{LP}^\eta$. We second note that the Jacobian constraint (35) is not directly enforced in our $\mathtt{LP}^\eta$. As discussed for a similar Jacobian constraint (14) in Proposition 1, a closely related Jacobian constraint $\|I - \mathbf{J}_N(\hat{\mathbf{u}}_N(\mu);\mu)^{-T}\tilde{\mathbf{J}}_N(\hat{\mathbf{u}}_N(\mu);\mu)^T\|_{\max}$ is LP admissible. However, we again omit the Jacobian constraint in our EQP formulation as (a) the inclusion of the Jacobian constraint would increase the number of manifold accuracy constraint per training parameter value from $3N$ to $3N + N^2$ and (b) in any event the effect of the Jacobian error on the output is second order (i.e., $\mathcal{O}(\delta_{\mathbf{J}}^2)$). Again, as will be shown in a numerical example in Section 5, we have found that the Jacobian constraint is not necessary in practice for aerodynamics problems tested.

4 | ADAPTATION

In this section we describe an algorithm that automatically trains the FE spaces, RB spaces, and EQP weights associated with the reduced model such that the model yields output predictions and error estimates that meet the user-prescribed tolerances. To this end, we first discuss the FE mesh adaptation strategy in Section 4.1. We next discuss the simultaneous FE, RB, and EQP training algorithm in Section 4.2. We then summarize the online computational procedure in Section 4.3.

4.1 | FE spatial adaptation

The FE adaptive solve follows the standard solve, estimate, mark, and refine strategy. The DG-FEM solver used in this work was described in Section 2.3. The error estimation for the quantity of interested is provided by the DWR method described in Section 3.1. To mark elements for refinement, we first localize the error estimate $\eta_h^{\rm fe}$ according to

$$\eta_\kappa^{\rm fe} \equiv |r_h(u_h(\mu), z_{\tilde h}(\mu)|_\kappa; \mu)| \quad \forall \kappa \in \mathcal{T}_h;$$

we then mark top 10% of the elements with the largest error indicators for refinement. We finally refine the marked element using the standard hanging-node refinement with the one-level regularity constraint. While our formulation works with any sequence of nested meshes, we consider quadrilateral and hexahedral meshes for two- and three-dimensional numerical examples, respectively, in Section 5.

4.2 | Simultaneous FE, RB, and EQP offline training

We now present our goal-oriented offline training algorithm, Algorithm 1, for the FE space, RB spaces, and EQP weights. The input to the algorithm are as follows: the parameter training set $\Xi_J^{\rm train} \subset \mathcal{D}$ consists of J parameter points; the FE output error tolerance $\delta^{\rm fe} \in \mathbb{R}_{>0}$; the RB output error tolerance $\delta^{\rm rb} \in \mathbb{R}_{>0}$; the three EQP tolerances δ^r, δ^q, and δ^η for the residual, output functional, and DWR, respectively; and the maximum dimension of the RB space. The output of the algorithm are as follows: the primal and dual reduced bases $\{\phi_i^{\rm pr}\}_{i=1}^{N_{\rm max}}$ and $\{\phi_i^{\rm du}\}_{i=1}^{N_{\rm max}}$, respectively; and the three sets of EQP weights $\{\rho_\kappa^r\}_{\kappa\in\mathcal{T}_h}$, $\{\rho_\kappa^q\}_{\kappa\in\mathcal{T}_h}$, and $\{\rho_\kappa^\eta\}_{\kappa\in\mathcal{T}_h}$ for residual, output functional, and DWR, respectively.

Before we describe the algorithm, we discuss the choice of FE, RB, and EQP tolerances. For the FE and RB tolerances, we recall that we wish to control the total output error $|s(\mu)-\tilde s_N(\mu)|$ for (at least) snapshot parameters $\mu \in \Xi_N^{\rm rb}$ to be less than a user-prescribed error tolerance say ϵ; since $|s(\mu)-\tilde s_N(\mu)| \leq |s(\mu)-s_h(\mu)|+|s_h(\mu)-\tilde s_N(\mu)| \lesssim \eta_h^{\rm fe}(\mu)+\tilde\eta_N^{\rm rb}(\mu) \lesssim \delta^{\rm fe}+\delta^{\rm rb}$, we select the sum of $\delta^{\rm fe}$ and $\delta^{\rm rb}$ to be less than the desired user-prescribed output error tolerance ϵ. For the EQP tolerances associated with output prediction (i.e., δ^r and δ^q), we choose $\delta^r \leq \delta^{\rm rb}$ and $\delta^q \leq \delta^{\rm rb}$ such that the EQP hyperreduction error is smaller than the RB approximation error. In practice, we choose $\delta^r \approx \delta^{\rm rb}/10$ and $\delta^q \approx \delta^{\rm rb}/100$; we choose a tight output functional tolerance δ^q because (i) the output functional EQP evaluation error $|q_h(\tilde u_N(\mu)) - \tilde q_h(\tilde u_N(\mu))|$ is not accounted for in our DWR formulation and (ii) the number of nonzero output functional EQP weights tends to be small even for a tight error tolerance. For the EQP tolerance δ^η associated with the output error estimate, we choose $\delta^\eta \approx \delta^{\rm rb}/10$; we choose the DWR EQP tolerance to be tighter than the RB tolerance $\delta^{\rm rb}$ so that the error estimate $\tilde\eta_N^{\rm rb}(\mu)$ remain effective even when the output error approaches $\delta^{\rm rb}$.

We now discuss Algorithm 1. Suppose we have constructed a reduced model of dimension $N-1$ and wish to construct a reduced model of dimension N. We first solve the output and the associated error estimates for all $\mu \in \Xi_J^{\rm train}$; we then choose the parameter value that yields the largest error estimate as the next snapshot parameter $\mu^{(N)}$ (line 6). We then solve for $u_h(\mu^{(N)})$ using the adaptive FE solver discussed in Section 4.1 so that $\eta_h^{\rm fe}(\mu^{(N)}) \leq \delta^{\rm fe}$ (line 9). If the FE space $\mathcal{V}_h$ is refined in the previous step, we then update the RBs $\{\phi_i^{\rm pr}\}_{i=1}^{N-1}$ and $\{\phi_i^{\rm du}\}_{i=1}^{N-1}$ by recomputing the snapshots $\{u_h(\mu)\}_{\mu\in\Xi_{N-1}^{\rm rb}}$ and $\{z_h(\mu)\}_{\mu\in\Xi_{N-1}^{\rm rb}}$ on the refined $\mathcal{V}_h$ and reorthonormalizing the RBs (line 10); although this step in principle should not be necessary since the original snapshots in the coarser space had met the tolerance $\delta^{\rm fe}$, we perform this step because (i) updating the snapshots for the refined mesh requires a small number of nonlinear solver iterations and (ii) we have found in practice the step improves the RB convergence. We next compute the dual FE solution $z_h(\mu^{(N)})$ (line 11). We then update the snapshot parameter set $\Xi_N^{\rm rb}$ and the associated primal and dual RBs $\{\phi_i^{\rm pr}\}_{i=1}^N$ and $\{\phi_i^{\rm du}\}_{i=1}^N$, respectively (line 12); the RBs are $\mathcal{V}$-orthonormalized using the Gram-Schmidt algorithm. We then update the residual EQP weights $\{\rho_\kappa^r\}$ (lines 13–16). We recall that $\texttt{LP}^r$, which finds the residual EQP weights, itself requires an approximate solutions $U_J^{\rm train} = \{\hat u_N(\mu)\}_{\mu\in\Xi_J^{\rm train}}$. To find these approximate solutions, we use a "bootstrapping" strategy which starts the training process using our current best approximation of the states. Namely, the training state $U_J^{\rm train} = \{\hat u_N(\mu)\}_{\mu\in\Xi_J^{\rm train}}$ comprises the FE solution for snapshot parameters $\mu \in \Xi_N^{\rm rb}$ and our current RB-EQP approximation for $\mu \in \Xi_J^{\rm train} \setminus \Xi_N^{\rm rb}$. We have found in practice $N_{\rm eqp,smooth} = 3$ is sufficient to obtain good EQP weights. We finally update the output functional and DWR EQP weights $\{\rho_\kappa^q\}$ and $\{\rho_\kappa^\eta\}$ (lines 18–19); note that, unlike in the computation of $\{\rho_\kappa^r\}$, we need not iterate here since the RB-EQP states, which are used as training states, do not depend on $\{\rho_\kappa^q\}$ and $\{\rho_\kappa^\eta\}$.

We now comment on the offline cost of the training algorithm. For a typical (nonlinear) aerodynamics problem, the dominant computational cost of the algorithm are (i) the adaptive computation of the FE solution (line 9) and (ii) the identification of the EQP weights (lines 15, 18, and 19). As regard (i), the adaptive computation of the FE solution requires the evaluation of (potentially) multiple FE solutions; the evaluation of each FE solution requires multiple Newton(-like) iterations, each of which

Algorithm 1 Greedy algorithm for goal-oriented simultaneous FE, RB, and EQP training.

```
Input:
    parameter training set: $\Xi_J^{\text{train}} \subset \mathcal{D}$
    FE tolerance: $\delta^{\text{fe}} \in \mathbb{R}_{>0}$
    RB tolerance: $\delta^{\text{rb}} \in \mathbb{R}_{>0}$
    EQP tolerances: $\delta^r, \delta^q, \delta^\eta \in \mathbb{R}_{\geq 0}$
    maximum RB space dimension: $N_{\max} \in \mathbb{R}_{>0}$
Output:
    reduced bases: $\{\phi_i^{\text{pr}}\}_{i=1}^{N_{\max}}$, $\{\phi_i^{\text{du}}\}_{i=1}^{N_{\max}}$
    EQP weights: $\{\rho_\kappa^r\}_{\kappa \in \mathcal{T}_h}$, $\{\rho_\kappa^q\}_{\kappa \in \mathcal{T}_h}$, $\{\rho_\kappa^\eta\}_{\kappa \in \mathcal{T}_h}$

1:  Set $\Xi_{N=0}^{\text{rb}} = \emptyset$, $\{\phi_i^{\text{pr}}\}_{i=1}^0 = \{\phi_i^{\text{du}}\}_{i=1}^0 = \emptyset$.
2:  for $N = 1, \dots, N_{\max}$ do
3:      if $N = 1$ then
4:          Set $\mu^{(N)} = \arg\inf_{\mu \in \Xi_J^{\text{train}}} \|\bar{\mu} - \mu\|$ where $\bar{\mu} \equiv \frac{1}{N} \sum_{\mu \in \Xi_J^{\text{train}}} \mu$.
5:      else
6:          Set $\mu^{(N)} = \arg\sup_{\mu \in \Xi_J^{\text{train}}} \tilde{\eta}_N^{\text{rb}}(\mu) \equiv \arg\sup_{\mu \in \Xi_J^{\text{train}}} |\tilde{r}_h(\tilde{u}_{N-1}(\mu), \tilde{z}_{N-1}(\mu); \mu)|$
7:      end if
8:      If $\tilde{\eta}_N^{\text{rb}}(\mu^{(N)}) < \delta^{\text{rb}}$, terminate.
9:      Find FE solution $u_h(\mu^{(N)}) \in \mathcal{V}_h$; invoke adaptivity as necessary such that $\eta_h^{\text{fe}}(\mu^{(N)}) \leq \delta^{\text{fe}}$.
10:     If $\mathcal{V}_h$ was refined in the previous step, then resolve $\{\phi_i^{\text{pr}}\}_{i=1}^{N-1}$ and $\{\phi_i^{\text{pr}}\}_{i=1}^{N-1}$ on $\mathcal{V}_h$.
11:     Find the dual FE solution $z_h(\mu^{(N)}) \in \mathcal{V}_h$.
12:     Update RB: $\Xi_N^{\text{rb}} = \Xi_{N-1}^{\text{rb}} \cup \mu^{(N)}$; $\{\phi_i^{\text{pr}}\}_{i=1}^N = \text{GS}_{\mathcal{V}}\{\{\phi_i^{\text{pr}}\}_{i=1}^{N-1}, u_h(\mu^{(N)})\}$;
                    $\{\phi_i^{\text{du}}\}_{i=1}^N = \text{GS}_{\mathcal{V}}\{\{\phi_i^{\text{du}}\}_{i=1}^{N-1}, z_h(\mu^{(N)})\}$.
13:     for $i = 1, \dots, N_{\text{eqp,smooth}}$ do
14:         Set $U_J^{\text{train}} \equiv \{\hat{u}_N(\mu)\}_{\mu \in \Xi_J^{\text{train}}}$ where $\hat{u}_N(\mu) = \begin{cases} u_h(\mu) & \forall \mu \in \Xi_N^{\text{rb}} \\ \tilde{u}_N(\mu) & \forall \mu \in \Xi_J^{\text{train}} \setminus \Xi_N^{\text{rb}} \end{cases}$.
15:         Solve $\text{LP}_N^r(\Xi_J^{\text{train}}, U_J^{\text{train}}, \delta^r)$ for $\{\rho_\kappa^r\}_{\kappa \in \mathcal{T}_h}$.
16:     end for
17:     Set $U_J^{\text{train}} \equiv \{\tilde{u}_N(\mu)\}_{\mu \in \Xi_J^{\text{train}}}$.
18:     Solve $\text{LP}_N^q(\Xi_J^{\text{train}}, U_J^{\text{train}}, \delta^q)$ for $\{\rho_\kappa^q\}_{\kappa \in \mathcal{T}_h}$.
19:     Solve $\text{LP}_N^\eta(\Xi_J^{\text{train}}, U_J^{\text{train}}, \delta^\eta)$ for $\{\rho_\kappa^\eta\}_{\kappa \in \mathcal{T}_h}$.
20: end for
```

requires one FE residual evaluation, one FE Jacobian evaluation, and one linear solve using an iterative solver. Hence, the cost of (i) per training iteration is $\mathcal{O}(\mathcal{N}^{\cdot})$, where the leading constant can be potentially large.

As regard (ii), the evaluation of the EQP weights — and in particular the residual weights $\{\rho_\kappa^r\}$ and the DWR weights $\{\rho_\kappa^\eta\}$ — requires the evaluation of the FE residual for all of J parameter values in Ξ_J^{train}; hence, the cost to evaluate the manifold accuracy constraints for LP^r and LP^η are $\mathcal{O}(J\mathcal{N})$. However, we note that the evaluation of a FE residual is significantly cheaper than the evaluation of a FE solution, as the residual evaluation does not require Newton(-like) iterations; hence the cost to evaluate $\mathcal{O}(J)$ EQP constraints is comparable to, or a small constant multiple of, the cost to evaluate a single FE solution (i.e., cost (i)). The evaluation of the manifold accuracy constraints for the output functional in LP^q is at least an order of magnitude smaller than those for LP^r and LP^η for typical aerodynamics problems. For all three LPs, the cost to solve the LP is a small fraction of the cost to evaluate the constraints.

The rest of the steps in the algorithms are relatively inexpensive. In particular, the greedy sampling of the output error is fast thanks to the online-efficient output error estimates (line 6). Similarly, the boostrapping strategy for the EQPs permits the use of online-efficient RB-EQP solutions as the EQP training states (lines 14 and 17). Hence, the overall cost of a single iteration of the training algorithm is comparable to, or at least a small constant multiple of, the cost to evaluate the FE solution for the single snapshot parameter value.

Remark 7. Our simultaneous FE, RB, and EQP offline training yields a single adaptively refined FE space for the entire parameter domain and represents all RB snapshots in this FE space. The use of a single "common" FE space significantly simplifies the implementation. However, this approach can be limiting in the presence of features (i) whose locations are strongly parameter dependent and (ii) that require aggressive mesh refinement, such as parameter-dependent shock waves. In such a problem, a "common" FE space could have significantly higher degrees of freedom than an FE space refined for a single parameter for a given error level. In addition, in such a problem the use of a single RB space itself can lead to slow convergence[31]. One approach to remedy the issue, though we do not pursue in this work, is to partition the parameter domain into subdomains to limit the range of the movement of the features[32].

4.3 | Online computational procedure

We now discuss the online storage requirement and computational procedure. In the offline stage, we invoke Algorithm 1 to simultaneously identify the FE space $\mathcal{V}_h$, the primal and dual RBs $\{\phi_i^{\mathrm{pr}} \in \mathcal{V}_h\}_{i=1}^N$ and $\{\phi_i^{\mathrm{du}} \in \mathcal{V}_h\}_{i=1}^N$, respectively, and the three sets of EQP weights $\{\rho_\kappa^r\}_{\kappa\in\mathcal{T}_h}$, $\{\rho_\kappa^q\}_{\kappa\in\mathcal{T}_h}$, and $\{\rho_\kappa^\eta\}_{\kappa\in\mathcal{T}_h}$ for residual, output functional, and DWR, respectively. We then identify the primal and dual EQP element sets

$$\tilde{\mathcal{T}}_h^{\mathrm{pr}} \equiv \{\kappa \in \mathcal{T}_h \mid \rho_{\kappa'}^r \neq 0, \rho_{\kappa'}^q \neq 0, \text{ or } \rho_{\kappa'}^\eta \neq 0 \text{ for } \kappa' = (\kappa \text{ or facet neighbor of } \kappa)\},$$
$$\tilde{\mathcal{T}}_h^{\mathrm{du}} \equiv \{\kappa \in \mathcal{T}_h \mid \rho_{\kappa'}^\eta \neq 0 \text{ for } \kappa' = (\kappa \text{ or facet neighbor of } \kappa)\}.$$

We then store (FE coefficients associated with) the primal and dual RBs restricted to the EQP element sets, $\{\{\phi_i^{\mathrm{pr}}|_\kappa\}_{\kappa\in\tilde{\mathcal{T}}_h^{\mathrm{pr}}}\}_{i=1}^N$ and $\{\{\phi_i^{\mathrm{du}}|_\kappa\}_{\kappa\in\tilde{\mathcal{T}}_h^{\mathrm{du}}}\}_{i=1}^N$. These restriction of the RBs, in conjunction with the EQP weights $\{\rho_\kappa^r\}_{\kappa\in\mathcal{T}_h}$, $\{\rho_\kappa^q\}_{\kappa\in\mathcal{T}_h}$, and $\{\rho_\kappa^\eta\}_{\kappa\in\mathcal{T}_h}$, store necessary and sufficient information to evaluate the EQP residual (6), the EQP output functional (7), and the EQP DWR error estimate (26) in the online stage. Note that online storage requirement for the primal and dual RBs are $N_{\mathrm{loc}}N|\tilde{\mathcal{T}}_h^{\mathrm{pr}}|$ and $N_{\mathrm{loc}}N|\tilde{\mathcal{T}}_h^{\mathrm{du}}|$, respectively, where N_{loc} is the number of FE degrees of freedom per element which depends on the polynomial degree p and the number of state components m. In particular, assuming $|\tilde{\mathcal{T}}_h^{\mathrm{pr}}| = \mathcal{O}(N)$ and $|\tilde{\mathcal{T}}_h^{\mathrm{du}}| = \mathcal{O}(N)$, the online storage requirement is $\mathcal{O}(N^2)$ and is independent of the underlying FE space.

We now discuss the online computational procedure. Given $\mu \in \mathcal{D}$, we first solve the RB-EQP problem (8),

$$\tilde{r}_h(\tilde{u}_N(\mu), v; \mu) = 0 \quad \forall v \in \mathcal{V}_N^{\mathrm{pr}},$$

for $\tilde{u}_N(\mu)$ using a Newton-like method. This step uses the RB-EQP residual form $\tilde{r}_h(\cdot, \cdot; \cdot)$, which is based on the EQP weights $\{\rho_\kappa^r\}_{\kappa\in\mathcal{T}_h}$ computed by $\mathtt{LP}^r$. This step requires multiple evaluation of the EQP residual and Jacobian as well as linear solves at the cost of $\mathcal{O}(N|\tilde{\mathcal{T}}_h^{\mathrm{pr}}|)$, $\mathcal{O}(N^2|\tilde{\mathcal{T}}_h^{\mathrm{pr}}|)$ and $\mathcal{O}(N^3)$, respectively.

Once we obtain the solution $\tilde{u}_N(\mu)$, we evaluate the EQP output given by (9),

$$\tilde{s}_N(\mu) \equiv \tilde{q}_h(\tilde{u}_N(\mu); \mu),$$

to provide an output prediction $\tilde{s}_N(\mu)$. This step uses the RB-EQP output functional form $\tilde{q}_h(\cdot; \cdot)$, which is based on the EQP weights $\{\rho_\kappa^q\}_{\kappa\in\mathcal{T}_h}$ computed by $\mathtt{LP}^q$. The evaluation of the form requires $\mathcal{O}(N|\tilde{\mathcal{T}}_h^{\mathrm{pr}}|)$ operations.

Finally, to provide the error estimate, we first compute the dual solution $\tilde{z}_N^{\mathrm{du}}(\mu) \in \mathcal{V}_N^{\mathrm{du}}$ given by (25),

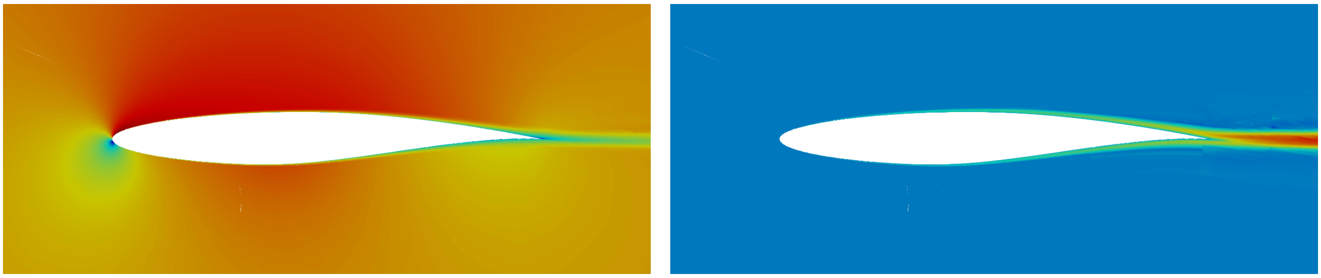
$$\tilde{r}_h'(\tilde{u}_N(\mu); v, \tilde{z}_N^{\mathrm{du}}(\mu); \mu) = \tilde{q}_h'(\tilde{u}_N(\mu); v; \mu) \quad \forall v \in \mathcal{V}_N^{\mathrm{du}}.$$

We then evaluate the output error estimate $\tilde{\eta}_N^{\mathrm{rb}}(\mu)$ given by (26),

$$\tilde{\eta}_N^{\mathrm{rb}}(\mu) \equiv |\tilde{r}_h(\tilde{u}_N(\mu), \tilde{z}_N^{\mathrm{du}}(\mu); \mu)|.$$

These two steps uses the forms associated with EQP weights $\{\rho_\kappa^\eta\}_{\kappa\in\mathcal{T}_h}$ computed by $\mathtt{LP}^\eta$. The steps require the assembly of the adjoint problem, the solution of the adjoint problem, and the evaluation of the DWR residual at the cost of $\mathcal{O}(N^2|\tilde{\mathcal{T}}_h^{\mathrm{du}}|)$, $\mathcal{O}(N^3)$, and $\mathcal{O}(N|\tilde{\mathcal{T}}_h^{\mathrm{du}}|)$, respectively.

In summary, assuming $|\tilde{\mathcal{T}}_h^{\mathrm{pr}}| = \mathcal{O}(N)$ and $|\tilde{\mathcal{T}}_h^{\mathrm{du}}| = \mathcal{O}(N)$, the online computational complexity to evaluate the output $\tilde{s}_N(\mu)$ and the associated error estimate $\tilde{\eta}_N^{\mathrm{rb}}(\mu)$ is $\mathcal{O}(N^3)$; the online complexity is independent of the dimension of the underlying FE space.



(a) Mach number

(b) SA working variable

FIGURE 1 Solution to the RAE 2822 problem for the centroidal parameter value of $\alpha = 2°$ and $M_\infty = 0.3$.

5 | EXAMPLES: PARAMETRIZED AERODYNAMICS PROBLEMS

In this section we consider two parametrized aerodynamics problems: two-dimensional turbulent flow over an RAE 2822 airfoil and three-dimensional turbulent flow over an ONERA M6 wing. The turbulent flows are modeled by the Reynolds-averaged Navier-Stokes (RANS) equations with the Spalart-Allmaras (SA) turbulence model[33]. Specifically, the flow equations are expressed in the entropy variables[34], and the SA equation is in the so-called SA-neg form[35]. The two input parameters that we consider are the angle of attack and the free stream Mach number; the specific parameter range will be subsequently described for each case. For both cases, compressibility effect is present but the flow remains subsonic (i.e., the maximum local Mach number remains below unity) for the entire parameter range so that there are no shocks in the flow. The mean-flow solutions of the RANS equations are hence smooth and do not contain parameter-dependent discontinuities. On the other hand, the SA turbulence model[33] exhibits a singular perturbation on the outer edge of the turbulent boundary layer whose location depends on the parameter value; however, this feature does not appear to cause convergence issues at least at the drag error level we consider in the two cases. Hence, we expect rapid convergence of the RB(-EQP) error for these cases.

5.1 | Two-dimensional turbulent flow over an RAE2822 airfoil

We consider two-dimensional turbulent flow over an RAE2822 airfoil modeled by the RANS equations with the SA turbulence model. We consider two input parameters: the angle of attack $\mu_1 \equiv \alpha \in [1°, 3°]$ and the free stream Mach number $\mu_2 \equiv M_\infty \in [0.2, 0.4]$. The Reynolds number is $Re_c = 6.5 \times 10^6$. The output of interest is the drag. The solution for the centroidal parameter value of $\alpha = 2°$ and $M_\infty = 0.3$ is shown in Figure 1. The initial mesh is the coarse mesh supplied for the first AIAA high-order workshop and comprises 506 elements. The training parameter set Ξ_J^{train} comprises $J = 5 \times 5$ uniformly distributed points in the parameter domain $\mathcal{D} \equiv [1°, 3°] \times [0.2, 0.4]$. (We consider a relatively small parameter domain; one approach to increase the parametric extent is to consider parameter-space decomposition as discussed in Remark 7.) The snapshots are computed using an isotropic h-adaptive $\mathbb{P}^2$ DG method. Over the parameter domain, the drag coefficient takes on a value in $[8.0 \times 10^{-3}, 9.1 \times 10^{-3}]$. Our goal is to construct a reduced model with a drag error of 10^{-4} (i.e., 1 drag count) or approximately 1%. To this end, we set the finite element and reduced basis output error tolerances to $\delta^{\text{fe}} = 2.5 \times 10^{-5}$ and $\delta^{\text{rb}} = 7.5 \times 10^{-5}$, respectively, and the EQP tolerances to $\delta^r = 10^{-5}$, $\delta^q = 10^{-6}$, and $\delta^\eta = 10^{-5}$. We refer to Section 4.2 for a discussion on the choices of the FE, RB, and EQP tolerances. The nonlinear problems associated with the FE, RB, and RB-EQP discretization are solved using a pseudo-time continuation (PTC) method, a Newton-like method designed to robustly solve highly nonlinear problems associated with aerodynamic flows; we refer to[36] for the particular PTC method used in this work.

Table 1(a) shows the convergence behavior of our goal-oriented model reduction method for the case where the test parameter set is the same as the training parameter set, $\Xi^{\text{test}} = \Xi_J^{\text{train}}$. We first comment on the dimension of the RB space and the number of nonzero EQP weights, shown in the first and second blocks of Table 1(a); these values directly affect the cost of the reduced model evaluation. The RB dimension of $N = 12$ is sufficient to meet the user-prescribed output error tolerance of $\delta^{\text{rb}} = 7.5 \times 10^{-5}$ in this case. We observe that the number of nonzero residual EQP weights $M^r \equiv \text{nnz}\{\rho_\kappa^r\}$ increases gradually with N; the number of nonzero weights of $M^r = 114$ for $N = 12$ is nevertheless less than 5% of the number of elements in the final adapted FE mesh with 2454 elements. We next observe that the number of nonzero output functional EQP weights $M^q \equiv \text{nnz}\{\rho_\kappa^q\}$ also increases with N; however, the number of nonzero output functional EQP weights is relatively small despite the use of tight

TABLE 1 Convergence behavior for the RAE 2822 problem. The columns are as follows: first column shows the dimension of the RB space; the next three columns show the number of nonzero EQP weights for residual $\{\rho_\kappa^r\}$, output functional $\{\rho_\kappa^q\}$, and DWR $\{\rho_\kappa^\eta\}$; the next two columns show the output error and output error estimate; the last three columns show the error due to EQP in the output, output functional, and error estimate. For fifth through ninth columns, the reported values are the maximum values over the test set Ξ^{test}; e.g., the output error reported is $\sup_{\mu\in\Xi^{\text{test}}} |s_h(\mu) - \tilde{s}_N(\mu)|$. Recall the EQP tolerances are $\delta^r = 10^{-5}$, $\delta^q = 10^{-6}$, and $\delta^\eta = 10^{-5}$.

(a) $\Xi^{\text{test}} = \Xi_N^{\text{train}}$

N	M^r	M^q	M^η	$\lvert s_h - \tilde{s}_N\rvert$	$\tilde{\eta}_N^{\text{rb}}$	$\lvert q(u_N) - q(\tilde{u}_N)\rvert$	$\lvert q(\tilde{u}_N) - \tilde{q}(\tilde{u}_N)\rvert$	$\lvert \eta_N^{\text{rb}} - \tilde{\eta}_N^{\text{rb}}\rvert$
1	4	5	11	2.69×10^{-2}	2.44×10^{-2}	3.15×10^{-5}	1.00×10^{-6}	7.58×10^{-6}
3	10	9	85	4.14×10^{-3}	3.09×10^{-3}	8.98×10^{-5}	1.00×10^{-6}	8.58×10^{-6}
5	32	11	125	1.14×10^{-2}	1.45×10^{-2}	9.83×10^{-5}	1.05×10^{-6}	6.86×10^{-6}
7	69	14	147	2.55×10^{-3}	2.56×10^{-3}	9.68×10^{-5}	1.12×10^{-6}	2.62×10^{-6}
9	78	14	116	1.01×10^{-3}	1.00×10^{-3}	3.71×10^{-5}	1.46×10^{-6}	5.66×10^{-6}
11	103	15	126	1.11×10^{-4}	1.14×10^{-4}	1.04×10^{-5}	2.01×10^{-6}	7.62×10^{-6}
12	114	15	136	6.96×10^{-5}	6.66×10^{-5}	4.90×10^{-5}	2.43×10^{-6}	1.01×10^{-5}

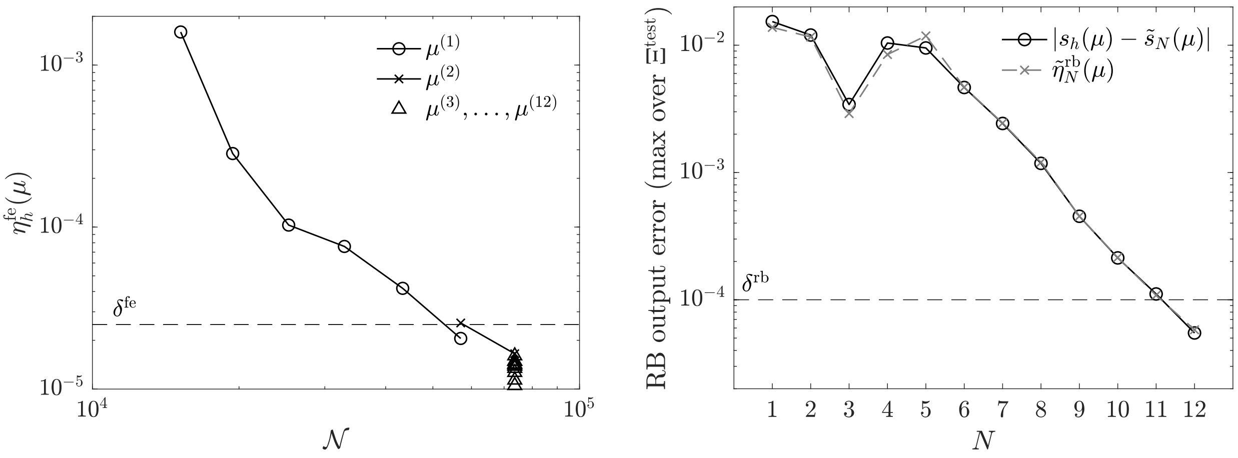
(b) $\Xi^{\text{test}} \neq \Xi_N^{\text{train}}$

N	M^r	M^q	M^η	$\lvert s_h - \tilde{s}_N\rvert$	$\tilde{\eta}_N^{\text{rb}}$	$\lvert q(u_N) - q(\tilde{u}_N)\rvert$	$\lvert q(\tilde{u}_N) - \tilde{q}(\tilde{u}_N)\rvert$	$\lvert \eta_N^{\text{rb}} - \tilde{\eta}_N^{\text{rb}}\rvert$
1	4	5	11	1.53×10^{-2}	1.38×10^{-2}	9.52×10^{-6}	8.84×10^{-7}	6.09×10^{-6}
3	10	9	85	3.42×10^{-3}	2.90×10^{-3}	7.26×10^{-5}	1.21×10^{-6}	1.67×10^{-5}
5	32	11	125	9.52×10^{-3}	1.18×10^{-2}	7.85×10^{-5}	1.05×10^{-6}	1.02×10^{-5}
7	69	14	147	2.43×10^{-3}	2.45×10^{-3}	8.20×10^{-5}	1.62×10^{-6}	2.50×10^{-6}
9	78	14	116	4.53×10^{-4}	4.53×10^{-4}	2.80×10^{-5}	1.97×10^{-6}	3.77×10^{-6}
11	103	15	126	1.11×10^{-4}	1.08×10^{-4}	1.99×10^{-5}	1.92×10^{-6}	7.12×10^{-6}
12	114	15	136	5.49×10^{-5}	5.81×10^{-5}	3.94×10^{-5}	2.34×10^{-6}	8.43×10^{-6}

output functional EQP tolerance $\delta^q = 10^{-6}$ because the output functional is a relatively simple integral compared to the residual. We also observe that the number of nonzero DWR EQP weights $M^\eta \equiv \text{nnz}\{\rho_\kappa^\eta\}$ increases with N; the number of nonzero DWR EQP weights is larger than that for the residual, but this does not cause a significant increase in the overall online evaluation cost because the DWR error estimate is performed only once per evaluation whereas the residual is evaluated multiple times in the Newton(-like) solver.

We next comment on the accuracy of the RB output predictions and the associated error estimates, shown in the third block of Table 1(a). The maximum RB output error $\max_{\mu\in\Xi_J^{\text{train}}} |s_h(\mu) - \tilde{s}_N(\mu)|$ converge rapidly with N, decreasing by over two orders of magnitude using just $N = 12$ RB. The maximum relative error for the $N = 12$ approximation is $\approx 0.9\%$. We also observe that the output RB-EQP error estimate $\tilde{\eta}_N^{\text{rb}}(\mu)$ provides an effective estimate of the output error. In particular, for $N \geq 7$, the error in the maximum error estimate is less than 5%. We observe that, by forgoing rigorous error *bounds* and accepting error *estimates*, the DWR error estimate eliminates the stability constant in more standard RB error estimates[3] or the Brezzi-Rappaz-Raviart formulation[16], and inasmuch also provides sharp estimates (albeit not bounds) for the convection-dominated problem. The effective error estimate implies that we can use the RB-EQP DWR error estimate $\tilde{\eta}_N^{\text{rb}}(\mu)$, which admits rapid $\mathcal{O}(N^{\cdot})$ evaluation, confidently in both offline greedy training and for online certification.

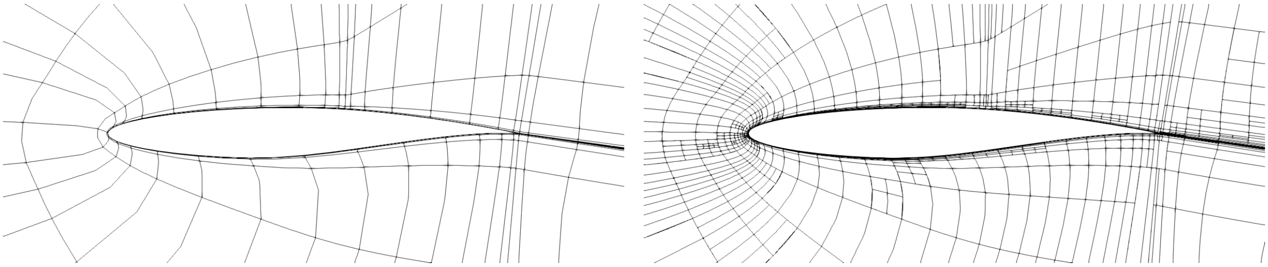
We finally comment on the accuracy of the EQP hyperreduction, shown in the fourth block of Table 1(a). We observe that the maximum error in the output due to the residual EQP, $\max_{\mu\in\Xi^{\text{test}}} |q_h(u_N(\mu);\mu) - q_h(\tilde{u}_N(\mu);\mu)|$, is of order of $\delta^r = 10^{-5}$; the residual EQP error is in fact least well-controlled out of the three EQP errors, most likely due to the strong nonlinearity present in the residual form associated with the RANS equations. We next observe that the maximum error in the output due to the output functional EQP, $\max_{\mu\in\Xi^{\text{test}}} |q_h(\tilde{u}_N(\mu);\mu) - \tilde{q}_h(\tilde{u}_N(\mu);\mu)|$, is again of order of $\delta^q = 10^{-6}$; the output functional EQP error is more tightly controlled than the residual counterpart, most likely as the EQP constraint (12) more directly relates to the error we wish to control. We finally observe that the maximum error in the DWR error estimate due to DWR EQP, $\max_{\mu\in\Xi^{\text{test}}} |\eta_N^{\text{rb}}(\mu) - \tilde{\eta}_N^{\text{rb}}(\mu)|$, is of order of $\delta^\eta = 10^{-5}$; the DWR EQP error, despite requiring both the adjoint and residual approximations, is relatively tightly controlled. The well-controlled EQP error in the DWR error estimate again ensures that we can use the online-efficient output



(a) FE error

(b) RB-EQP Error

FIGURE 2 The FE and RB-EQP error convergence for the RAE 2822 problem.



(a) initial mesh

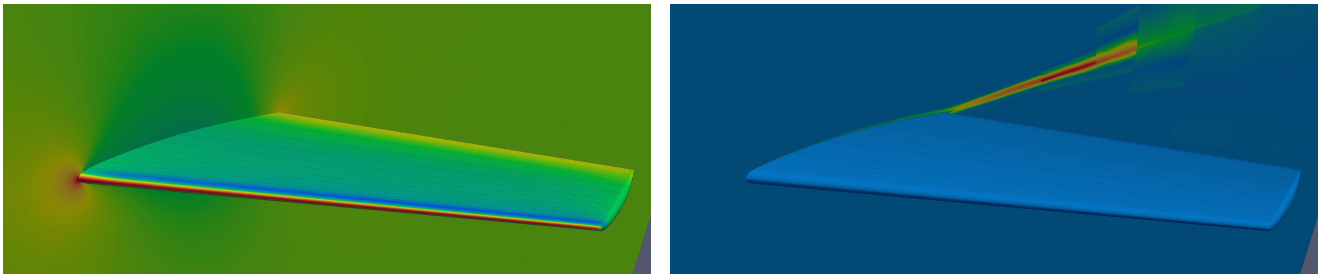
(b) final adapted mesh

FIGURE 3 Initial and final meshes for the RAE 2822 problem.

error estimate $\tilde{\eta}_N^{\mathrm{rb}}(\mu)$ confidently. Figure 2(b) summarized the rapid convergence of the RB-EQP output and the effectiveness of the DWR EQP error estimate.

Having discussed the convergence of the output error due to RB and EQP, we now discuss the convergence of the FE error in snapshots. Figure 2(a) shows the convergence of the output error estimate for all snapshots used in the reduced model. We observe that the evaluation of the drag on the initial coarse mesh with 506 elements ($\mathcal{N} = 15180$ for $\mathbb{P}^2$ approximation of the RANS equations with $m = 5$ components) shown in Figure 3(a) results in a drag error of over $1.5{\times}10^{-3}$ or 15% for $s_h(\mu^{(1)})$. The adaptive FE method rapidly decreases the error to less than the user-prescribed output tolerance of $\delta^{\mathrm{fe}} = 2.5 \times 10^{-5}$ in five adaptation iterations; the adapted mesh contains 1901 elements ($\mathcal{N} = 57030$). In the next greedy training iteration, however, the FE solver finds this mesh is insufficient for the accurate evaluation of $s_h(\mu^{(2)})$ and adaptively refines the mesh one more time to meet the user-prescribed output tolerance δ^{fe}; the new mesh contains 2454 elements ($\mathcal{N} = 73620$). This mesh, shown in Figure 3(b), is found to be sufficiently refined to provide accurate drag predictions for all subsequent parameter values $\mu^{(3)}, \dots, \mu^{(12)}$.

Table 1(b) shows the convergence behavior of the goal-oriented model reduction for the case where the test parameter set is a set of 25 uniformly distributed random points in $\mathcal{D}$ and in particular $\Xi^{\mathrm{test}} \neq \Xi_J^{\mathrm{train}}$. We find that the convergence behavior for this "predictive" case, where the test points are different from the training points, is very similar to the "reproduction" case shown in Table 1(a) due to the use of the sufficiently large training set Ξ_J^{train} for this problem. We note that the use of sufficiently large training set Ξ_J^{train} is afforded by the simultaneous FE, RB, and EQP training algorithm discussed in Section 4.2 which only requires FE calculation for the final snapshot parameter set $\Xi_N^{\mathrm{rb}} \subset \Xi_J^{\mathrm{train}}$. (The "predictive" case has smaller maximum errors than the "reproduction" case because the random Ξ^{test} has fewer points near the parameter domain boundary than the structured Ξ_J^{train}, where the maximum error is often encountered.)



(a) pressure

(b) SA working variable

FIGURE 4 Solution to the ONERA M6 problem for the centroidal parameter value of $\alpha = 1°$ and $M_\infty = 0.4$.

We finally report the online computational savings. All solvers are implemented in an in-house C++ code, and all computations are performed on a commodity desktop. Over 25 test cases defined by Ξ^{test}, the DG-RB-EQP solver ($N = 12$) on average takes $\approx 0.04 t_{\text{fe}}$ to provide an output prediction and $\approx 0.013 t_{\text{fe}}$ to provide the associated error estimate, where t_{fe} is the time to solve an FE problem on the mesh that has been already adapted (without FE error estimation). In other words, the DG-RB-EQP solver ($N = 12$) on average reduces wall-clock time by ≈ 25 for output prediction (only) and ≈ 20 for output prediction and error estimate. We note that these computational savings are relative to the adaptive high-order DG method; for the same error level, much greater savings would be observed relative to a lower-order and/or non-adaptive method.

5.2 | Three-dimensional turbulent flow over an ONERA M6 wing

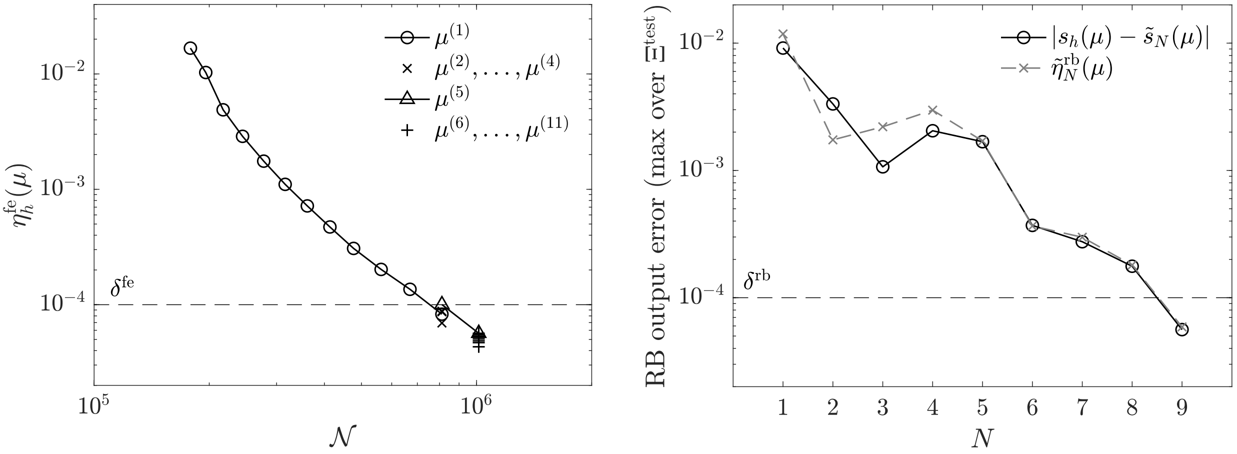
We now consider three-dimensional turbulent flow over an ONERA M6 wing modeled by the RANS equations with the SA turbulence model. A more detailed description of the RANS-SA equations is provided in the beginning of Section 5. We consider two input parameters: the angle of attack $\mu_1 \equiv \alpha \in [0°, 2°]$ and the free stream Mach number $\mu_2 \equiv M_\infty \in [0.3, 0.5]$. The Reynolds number is fixed at $Re_c = 10^6$. The output of interest is the drag. The solution for the centroidal parameter value of $\alpha = 1°$ and $M_\infty = 0.4$ is shown in Figure 4. The initial mesh is a coarse mesh that comprises 2976 elements. As in the RAE case, the training parameter set Ξ_J^{train} comprises $J = 5 \times 5$ uniformly distributed points over the parameter domain $\mathcal{D} \equiv [0°, 2°] \times [0.3, 0.5]$. The snapshots are computed using an anisotropic h-adaptive $\mathbb{P}^2$ DG method. Over the parameter domain, the drag coefficient takes on a value in $[2.03 \times 10^{-2}, 2.47 \times 10^{-2}]$. We wish to construct a reduced model with a drag error of 2×10^{-4} or approximately 1%. To this end, we set the finite element and reduced basis output error tolerances to $\delta^{\text{fe}} = \delta^{\text{rb}} = 10^{-4}$; the EQP tolerances are $\delta^{\text{r}} = 10^{-5}$, $\delta^{\text{q}} = 10^{-6}$, and $\delta^{\eta} = 10^{-5}$.

Table 2 shows the convergence behavior of the goal-oriented model reduction method for the "predictive" case where the test parameter is a set of 25 uniformly distributed random points in $\mathcal{D}$ and $\Xi^{\text{test}} \neq \Xi_J^{\text{train}}$. As the behavior for this ONERA M6 problem is similar to the RAE problem, we highlight the key figures and refer to Section 5.1 for more detailed discussions. The RB space dimension of $N = 9$ is sufficient to meet $\delta^{\text{rb}} = 10^{-4}$; the maximum output error is $\max_{\mu \in \Xi_J^{\text{test}}} |s_h(\mu) - \tilde{s}_N(\mu)| = 5.62 \times 10^{-5}$, which is the relative error of $\approx 0.3\%$. We obtain $M^r = 72$ nonzero residual EQP weights for $N = 9$, which is less than 0.5% of 16887 elements in the final adapted FE mesh; despite the significant reduction in the number of elements, the output error due to the EQP approximation $|q(u_N) - q(\tilde{u}_N)|$ is bounded by 6.5×10^{-5}, which is comparable to the target value of $\delta^r = 10^{-5}$. The number of output functional EQP elements M^q is small for all N; nevertheless, the error in the output functional evaluation due to EQP $|q(\tilde{u}_N) - \tilde{q}_N(\tilde{u}_N)|$ is bounded by 2.6×10^{-6}, which is comparable to the target value of $\delta^q = 10^{-6}$. Finally, the maximum number of DWR EQP elements is $M^\eta = 112$, which is greater than M^r but nevertheless less than 0.7% of the elements in the final adapted mesh; the error in the error estimate due to EQP $|\eta_N^{\text{rb}} - \tilde{\eta}_N^{\text{rb}}|$ is bounded by 6.1×10^{-6} which is comparable to the target value of $\delta^\eta = 10^{-6}$. Thanks to the control of each part of the total output error, overall the error between the adaptive FE and RB-EQP approximation $|s_h - \tilde{s}_N|$ decays rapidly with N, and the error estimate $\tilde{\eta}_N^{\text{rb}}$ provides a sharp estimate of the error. Figure 5(b) summarizes the performance of our goal-oriented model reduction method; similarly to the RAE problem, the RB-EQP output convergences rapidly and the DWR EQP error estimate is effective.

We now discuss the convergence of the FE error in the snapshots. Figure 5(a) shows the convergence of the output error estimate for all snapshots used in the reduced model. We observe that the evaluation of the drag on the initial coarse mesh with 2976 elements ($\mathcal{N} = 178560$ for $\mathbb{P}^2$ approximation of the RANS equations with $m = 6$ components) shown in Figure 6(a)

TABLE 2 Convergence behavior for the ONERA M6 problem with $\Xi^{\text{test}} \neq \Xi_J^{\text{train}}$. The columns are as follows: first column shows the dimension of the RB space; the next three columns show the number of nonzero EQP weights for residual $\{\rho_\kappa^r\}$, output functional $\{\rho_\kappa^q\}$, and DWR $\{\rho_\kappa^\eta\}$; the next two columns show the output error and output error estimate; the last three columns show the error due to EQP in the output, output functional, and error estimate. For fifth through ninth columns, the reported values are the maximum values over the test set Ξ^{test}; e.g., the output error reported is $\sup_{\mu \in \Xi^{\text{test}}} |s_h(\mu) - \tilde{s}_N(\mu)|$. Recall the EQP tolerances are $\delta^r = 10^{-5}$, $\delta^q = 10^{-6}$, and $\delta^\eta = 10^{-5}$.

N	M^r	M^q	M^η	$\lvert s_h - \tilde{s}_N \rvert$	$\tilde{\eta}_N^{\text{rb}}$	$\lvert q(u_N) - q(\tilde{u}_N) \rvert$	$\lvert q(\tilde{u}_N) - \tilde{q}(\tilde{u}_N) \rvert$	$\lvert \eta_N^{\text{rb}} - \tilde{\eta}_N^{\text{rb}} \rvert$
1	4	5	12	9.15×10^{-3}	1.18×10^{-2}	9.73×10^{-6}	1.02×10^{-6}	6.05×10^{-6}
3	7	10	61	1.07×10^{-3}	2.20×10^{-3}	6.31×10^{-6}	1.20×10^{-6}	5.13×10^{-6}
5	21	12	86	1.68×10^{-3}	1.70×10^{-3}	5.97×10^{-5}	1.19×10^{-6}	5.30×10^{-6}
7	47	13	111	2.75×10^{-4}	2.99×10^{-4}	6.22×10^{-5}	1.26×10^{-6}	4.85×10^{-6}
9	72	16	112	5.62×10^{-5}	5.90×10^{-5}	6.75×10^{-6}	2.57×10^{-6}	5.32×10^{-6}



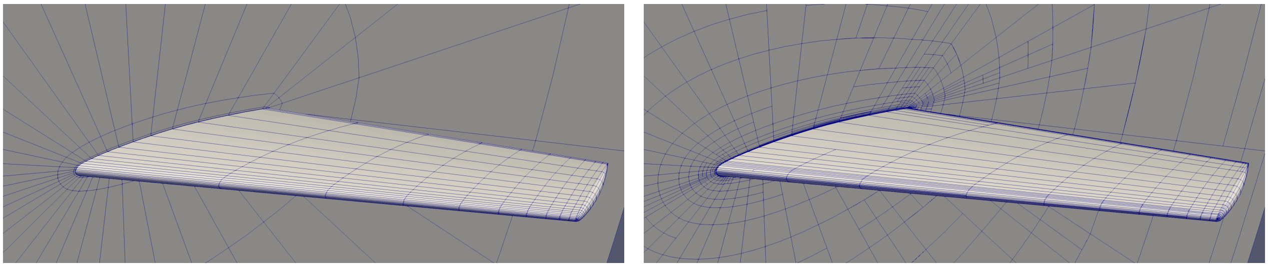
(a) FE error

(b) RB-EQP error

FIGURE 5 The FE and RB-EQP error convergence for the ONERA M6 problem.

results in a drag error of over 230% for $s_h(\mu^{(1)})$. The adaptive FE method rapidly decreases the error to less than the user-prescribed output tolerance of $\delta^{\text{fe}} = 1 \times 10^{-4}$; the adapted mesh shows significant refinement in the boundary layer and contains 13528 elements ($\mathcal{N} = 811680$). This mesh is used for $\mu^{(2)}, \ldots, \mu^{(4)}$, but a further refinement is performed for $\mu^{(5)}$; the refined mesh contains 16887 elements ($\mathcal{N} = 1013220$). This mesh, shown in Figure 6(b), is used for all subsequent parameter values $\mu^{(6)}, \ldots, \mu^{(9)}$.

We finally report the online computational savings; all computations are performed on a computational cluster with 80 cores. We use the CPU time required to solve an FE problem on the mesh that has been already adapted (without FE error estimation) as the time unit t_{fe}; we note that the cost is in CPU time and not wall-clock time. Over 25 test cases defined by Ξ^{test}, the DG-RB-EQP solver ($N = 9$) on average takes $\approx 0.0029 t_{\text{fe}}$ to provide an output prediction and $\approx 0.0005 t_{\text{fe}}$ to provide the associated error estimate. In other words, the DG-RB-EQP solver ($N = 9$) on average reduces CPU time by $\approx$ 340 for output prediction (only) and $\approx$ 290 for output prediction and error estimate. We again note that these computational savings are relative to the adaptive high-order DG method; for the same error level, much greater savings would be observed relative to a lower-order and/or non-adaptive method.



(a) initial mesh

(b) final adapted mesh

FIGURE 6 Initial and final meshes for the ONERA M6 problem.

ACKNOWLEDGMENTS

I would like to thank Prof. Anthony Patera (MIT) for many fruitful discussions. The financial support for this work was provided by the Natural Sciences and Engineering Research Council of Canada. 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.

I have no conflicts of interest.

References

1. Cockburn B. Discontinuous Galerkin methods. *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik* 2003; 83(11): 731–754.

2. Arnold DN, Brezzi F, Cockburn B, Marini LD. Unified analysis of discontinuous Galerkin methods for elliptical problems. *SIAM J. Numer. Anal.* 2002; 39(5): 1749–1779.

3. Rozza G, Huynh DBP, Patera AT. 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.* 2008; 15(3): 229–275.

4. Hesthaven JS, Rozza G, Stamm B. *Certified Reduced Basis Methods for Parametrized Partial Differential Equations*. Springer . 2016.

5. Barrault M, Maday Y, Nguyen NC, Patera AT. An “empirical interpolation” method: application to efficient reduced-basis discretization of partial differential equations. *C. R. Acad. Sci. Paris, Ser. I* 2004; 339: 667-–672.

6. Ryckelynck D. A priori hyperreduction method: an adaptive approach. *J. Comput. Phys.* 2005; 202(1): 346–366.

7. An SS, Kim T, James DL. Optimizing cubature for efficient integration of subspace deformations. *ACM Trans. Graph.* 2008; 27(5): 165:1–165:10.

8. Carlberg K, Bou-Mosleh C, Farhat C. Efficient non-linear model reduction via a least-squares Petrov-Galerkin projection and compressive tensor approximations. *Int. J. Numer. Methods Eng.* 2011; 86(2): 155–181.

9. Farhat C, Chapman T, Avery P. Structure-preserving, stability, and accuracy properties of the energy-conserving sampling and weighting method for the hyper reduction of nonlinear finite element dynamic models. *Int. J. Numer. Methods Eng.* 2015; 102(5): 1077–1110. nme.4820.

10. Patera AT, Yano M. An LP empirical quadrature procedure for parametrized functions. *C. R. Acad. Sci. Paris, Ser. I* 2017.

11. Yano M, Patera AT. An LP empirical quadrature procedure for reduced basis treatment of parametrized nonlinear PDEs. *Comput. Methods Appl. Mesh. Eng.* 2018.

12. Becker R, Rannacher R. An optimal control approach to a posteriori error estimation in finite element methods. *Acta Numerica* 2001; 10: 1–102.

13. Fidkowski KJ, Darmofal DL. Review of output-based error estimation and mesh adaptation in computational fluid dynamics. *AIAA journal* 2011; 49(4): 673–694.

14. Meyer M, Matthies HG. Efficient model reduction in non-linear dynamics using the Karhunen-Loève expansion and dual-weighted-residual methods. *Computational Mechanics* 2003; 31(1-2): 179–191.

15. Carlberg K. Adaptive *h*-refinement for reduced-order models. *Int. J. Numer. Methods Eng.* 2014; 102(5): 1192–1210.

16. Veroy K, Patera AT. Certified real-time solution of the parametrized steady incompressible Navier-Stokes equations: rigorous reduced-basis *a posteriori* error bounds. *Int. J. Numer. Methods Fluids* 2005; 47: 773–788.

17. Ali M, Steih K, Urban K. Reduced basis methods based upon adaptive snapshot computations. *Advances in Computational Mathematics* 2017; 43: 257–294.

18. Yano M. A reduced basis method for coercive equations with an exact solution certificate and spatio-parameter adaptivity: energy-norm and output error bounds. *SIAM J. Sci. Comput.* 2018; 40(1): A388–A420.

19. LeGresley P, Alonso J. Airfoil design optimization using reduced order models based on proper orthogonal decomposition. In: No. 2000-2545. AIAA Fluids 2000 Conference and Exhibit. ; 2000.

20. LeGresley PA, Alonso JJ. Investigation of non-linear projection for POD based reduced order models for aerodynamics. In: No. 2001–0926. 39th AIAA Aerospace Sciences Meeting & Exhibit. ; 2001.

21. LeGresley PA, Alonso JJ. Dynamic domain decomposition and error correction for reduced order models. In: No. 2003-0250. 41st AIAA Aerospace Sciences Meeting & Exhibit. ; 2003.

22. Washabaugh K, Zahr MJ, Farhat C. On the use of discrete nonlinear reduced-order models for the prediction of steady-state flows past parametrically deformed complex geometries. In: No. 2016-1814. 54th AIAA Aerospace Sciences Meeting. ; 2016.

23. Zimmermann R, Görtz S. Non-linear reduced order models for steady aerodynamics. *Procedia Computer Science* 2012; 1(1): 165–174. ICCS 2010.

24. Zimmermann R, Görtz S. Improved extrapolation of steady turbulent aerodynamics using a non-linear POD-based reduced order model. *The Aeronautical Journal* 2012; 116(1184): 1079–1100.

25. Bassi F, Rebay S. GMRES discontinuous Galerkin solution of the compressible Navier-Stokes equations. In: Cockburn B, Karniadakis GE, Shu CW., eds. *Discontinuous Galerkin Methods: Theory, Computation and Applications*Berlin: Springer. 2000 (pp. 197–208).

26. Bassi F, Crivellini A, Rebay S, Savini M. Discontinuous Galerkin solution of the Reynolds-averaged Navier-Stokes and *k*-*ω* turbulence model equations. *Computers & Fluids* 2005; 34(4-5): 507–540.

27. Yano M. Discontinuous Galerkin reduced basis empirical quadrature procedure for model reduction of parametrized nonlinear conservation laws. *Adv. Comput. Math.* 2019, accepted.

28. Hesthaven JS, Warburton T. *Nodal discontinuous Galerkin methods*. Springer New York . 2008.

29. Pietro DAD, Ern A. *Mathematical aspects of discontinuous Galerkin methods*. Springer Berlin Heidelberg . 2012.

30. Oliver TA, Darmofal DL. Analysis of dual consistency for discontinuous Galerkin discretizations of source terms. *SIAM J. Numer. Anal.* 2009; 47(5): 3507–3525.

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

32. Eftang JL, Patera AT, Rønquist EM. An "*hp*" certified reduced basis method for parametrized elliptic partial differential equations. *SIAM Journal on Scientific Computing* 2010; 32(6): 3170–3200.

33. Spalart PR, Allmaras SR. A one-equation turbulence model for aerodynamics flows. *La Recherche Aérospatiale* 1994; 1: 5–21.

34. Barth TJ. Numerical methods for gasdynamic systems on unstructured meshes. In: Kröner D, Ohlberger M, Rohde C., eds. *An Introduction to Recent Developments in Theory and Numerics for Conservation Laws*Springer-Verlag. 1999 (pp. 195–282).

35. Allmaras SR, Johnson FT, Spalart PR. Modifications and clarifications for the implementation of the Spalart-Allmaras turbulence model. 7th International Conference on Computational Fluid Dynamics ICCFD7-1902, ICCFD; : 2012.

36. Yano M, Modisette JM, Darmofal D. The importance of mesh adaptation for higher-order discretizations of aerodynamic flows. AIAA 2011-3852, AIAA; : 2011.

How to cite this article: M. Yano. (2020), Goal-oriented model reduction of parametrized nonlinear PDEs; application to aerodynamics *Int. J. Numer. Meth. Eng. 2020;00:1–6*.