INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING
Int. J. Numer. Meth. Engng 2008; **76**:27–55
Published online 25 February 2008 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/nme.2309

An efficient reduced-order modeling approach for non-linear parametrized partial differential equations

N. C. Nguyen*,† and J. Peraire

Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA 20139, U.S.A.

SUMMARY

For general non-linear parametrized partial differential equations (PDEs), the standard Galerkin projection is no longer efficient to generate reduced-order models. This is because the evaluation of the integrals involving the non-linear terms has a high computational complexity and cannot be pre-computed. This situation also occurs for linear equations when the parametric dependence is nonaffine. In this paper, we propose an efficient approach to generate reduced-order models for large-scale systems derived from PDEs, which may involve non-linear terms and nonaffine parametric dependence. The main idea is to replace the non-linear and nonaffine terms with a coefficient-function approximation consisting of a linear combination of pre-computed basis functions with parameter-dependent coefficients. The coefficients are determined efficiently by an inexpensive and stable interpolation at some pre-computed points. The efficiency and accuracy of this method are demonstrated on several test cases, which show significant computational savings relative to the standard Galerkin projection reduced-order approach. Copyright © 2008 John Wiley & Sons, Ltd.

Received 11 May 2007; Revised 18 December 2007; Accepted 20 December 2007

KEY WORDS: parametrized PDEs; reduced-order approximation; standard Galerkin reduced-order model; coefficient-function approximation; elliptic equations; convection–diffusion equations

1. INTRODUCTION

Many systems/processes in engineering and science are described by *parametrized* partial differential equations (PDEs). Typically, the quantities of engineering interests are not the full field variables, but rather certain *outputs*, best articulated as functionals of the field variables. Typical outputs include flow rate, pressure drops, concentration and flux, critical stresses or maximum displacements, and lift and drag forces. These outputs are functions of system parameters, or

*Correspondence to: N. C. Nguyen, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA 20139, U.S.A.
†E-mail: cuongng@mit.edu

Contract/grant sponsor: Singapore-MIT Alliance

inputs, that serve to identify a particular configuration of the system—geometry, material properties, initial and boundary conditions, and loads. The relevant system behavior is thus described by an implicit input–output relationship, evaluation of which demands solution of the underlying PDE. The design, optimization, control, and characterization of engineering systems require repeated and real-time output prediction.

Our goal is to develop an efficient reduced-order modeling approach for the rapid prediction of functional outputs associated with parametrized PDEs. Our approach is based on past and recent developments of the reduced-basis approaches (RBAs) [1–7] and reduced-order modeling techniques [8–14]. The fundamental observation is that the field variable is not an arbitrary member of the infinite-dimensional solution space associated with the underlying PDE; rather, it resides on a much lower-dimensional manifold induced by the parametric dependence. Low-order methods explicitly recognize and exploit this observation to develop reduced-order modeling of parametrized PDEs, often at a reduction of several orders of magnitude in the degrees of freedom compared with the classical models generated by finite element (FE), finite volume (FV), or finite difference (FD) methods.

In particular, the reduced-basis method has been successfully developed for (a) parametrized linear elliptic and parabolic PDEs that are *affine*[‡] in the parameter [7, 15–19] and (b) non-linear PDEs that are *at most quadratically non-linear* in the field variable [20–22]. In these cases, a very efficient reduced-basis approximation can be developed by exploiting the affine decomposition of the parametrized differential operator and resolving non-linear terms into the sum of products of the basis functions and coefficients; in such cases, the reduced-basis method offers significant—several orders of magnitude—computational savings relative to classical solution methods. However, while the assumptions of affine parameter dependence and quadratic non-linearity are crucial to computational efficiency, they restrict the application of the reduced-basis method to some specific domains. For example, problems involving nonaffine parameter dependence and highly non-linear terms do not fall into categories (a) and (b). Hence, when applied to such problems, the traditional reduced-basis method no longer provides efficiency to compete with classical solution methods.

The proper orthogonal decomposition (POD)-Galerkin approach has been widely used to produce reduced-order modeling of time-dependent PDEs. Nevertheless, its success and thus applications remain very much limited to linear and quadratically non-linear problems such as linearized Euler and parabolic equations [12, 13, 23], Burgers equation [10, 24], and incompressible Navier–Stokes equations [11, 25, 26]. Other model reduction techniques have also been developed for non-linear time-dependent problems. In particular, linearization methods [27–29] and polynomial approximation methods [30, 31] have been proposed to treat certain (weakly) non-linear problems quite satisfactorily. However, inefficient representation of the non-linear terms and fast growth of the computational complexity in the presence of strong non-linearity render these methods impractical for more general applications.

The classes of parametrized PDEs we consider here include (i) linear elliptic equations with *nonaffine parameter dependence*, (ii) *non-linear* elliptic equations, and (iii) *non-linear time-dependent* convection–diffusion equations. For such classes of PDEs, the reduced-order modeling provided by the standard Galerkin projection is no longer efficient. This is because the evaluation of the integrals involving the nonaffine and non-linear terms has a high computational complexity and cannot be pre-computed. To recover efficiency, we replace the nonaffine and non-linear terms

[‡] 'Affine parameter dependence' means that the parametrized differential operator can be expressed as a sum of products of *parameter-dependent* functions and *parameter-independent* operators [7, 15–17].

DOI: 10.1002/nme

with a *coefficient-function approximation* consisting of a linear combination of pre-computed basis functions and parameter-dependent coefficients. The coefficients are determined very efficiently by an inexpensive and stable interpolation at some pre-computed points. This allows us to apply an offline/online computational decomposition [7, 17, 21] for the generation and simulation of the reduced-order model. Unlike the standard Galerkin reduced-order approach (ROA), our method has a low computational complexity for the (online) evaluation of the integrals associated with the nonaffine and non-linear terms. The method can thus effect considerable computational economies relative to both the standard Galerkin ROA and classical solution methods.

Our work is in some aspects related to a new RBA, first introduced in [32] for linear elliptic problems with nonaffine parameter dependence and recently extended more broadly to non-linear elliptic and parabolic problems [33]. In particular, both the present approach and the approach discussed in the previous work [32, 33] deal with non-linearities by developing a coefficient-function approximation for the non-linear terms. The formulation and analysis of the resulting reduced model is thus similar for the two approaches. Although our paper here and [32, 33] address common issues in reduced-order modeling of non-linear problems, our method differs from these earlier efforts in two important aspects. First, rather than a greedy basis construction approach, we employ the POD approach to construct our low-dimensional approximation spaces. The POD basis provides the optimal representation of the given snapshot set in the mean square error sense and works well for time-dependent problems. Second, rather than the empirical interpolation method (EIM) [32, 33], we use the so-called *best points interpolation method* (BPIM) first introduced in [34] to develop coefficient-function expansions for the nonaffine and non-linear terms. Compared with the EIM, the BPIM is more expensive in the construction of interpolation points and basis functions, but provides approximations with higher accuracy than the EIM. As a result, with the BPIM we can use a smaller number of basis functions to achieve the same accuracy, which in turn leads to a more economical reduced-order model than with the EIM.

This paper is organized as follows. In Section 2, we present a short review of the BPIM. In subsequent sections, we introduce the abstract formulation and develop the reduced-order approximation for linear elliptic problems in Section 3, non-linear elliptic problems in Section 4, and non-linear time-dependent convection–diffusion problems in Section 5. Numerical examples are presented in each section in order to assess the efficiency and accuracy of our approach. Finally, in Section 6, we present concluding remarks.

2. BEST POINTS INTERPOLATION METHOD

In this section, we briefly describe the BPIM to construct a 'coefficient-function' approximation of parametrized functions. The approximation of such a class of functions is important for the effective reduced-order treatment of nonaffine and non-linear PDEs. We also refer the reader to [32, 33, 35] for an alternative approach—the EIM—in approximating parametrized fields. In addition to their use in non-linear model reduction, these methods can be applied in a variety of applications, for example, image or data compression involving domains of irregular profile, fast rendering and visualization in animation, the development of computer simulation surrogates or experimental response surface for design and optimization, and the determination of a good numerical integration scheme for smooth functions on irregular domains. See [34] for application of the BPIM to face recognition and optimal sensor placement for field reconstruction and [35] for application of the EIM to polynomial interpolation and numerical integration.

DOI: 10.1002/nme

2.1. Coefficient-function procedure

We consider the problem of approximating a given $\boldsymbol{\mu}$-dependent function $\mathbf{x} \to g(\mathbf{x};\boldsymbol{\mu}) \in L^\infty(\Omega) \cap C^0(\Omega)$, for all $\boldsymbol{\mu} \in \mathscr{D}$, by a coefficient-function expansion $g_M(\mathbf{x};\boldsymbol{\mu})$; here $\mathbf{x} = (x^1, \ldots, x^d)$ is a point of the physical domain $\Omega \in \mathbb{R}^d$, and $\mathscr{D} \in \mathbb{R}^P$ is the parameter space in which our P-tuple parameter vector $\boldsymbol{\mu} = (\mu^1, \ldots, \mu^P)$ resides. Toward this end, we assume that we are given an approximation space spanned by M orthonormal basis functions, $\Phi_M = \text{span}\{\phi_1, \ldots, \phi_M\}$, with $(\phi_i, \phi_j) = \delta_{ij}$, $1 \leqslant i, j \leqslant M$; here δ is the Kronecker symbol and $(\cdot,\cdot)$ denotes the $L^2(\Omega)$ inner product with an induced norm $\|\cdot\| = \sqrt{(\cdot,\cdot)}$. We further assume that we are given an associated set of M interpolation points $\{\mathbf{z}_m \in \Omega\}_{m=1}^M$.

Next, we define our coefficient-function approximation as

$$g_M(\mathbf{x};\boldsymbol{\mu}) = \sum_{m=1}^{M} \beta_{Mm}(\boldsymbol{\mu})\phi_m(\mathbf{x}) \tag{1}$$

where the coefficient vector $\boldsymbol{\beta}_M(\boldsymbol{\mu})$ is the solution of

$$\sum_{m=1}^{M} \phi_m(\mathbf{z}_i)\beta_{Mm}(\boldsymbol{\mu}) = g(\mathbf{z}_i;\boldsymbol{\mu}), \quad i = 1, \ldots, M \tag{2}$$

We observe from (1)–(2) that $g_M(\mathbf{x};\boldsymbol{\mu})$ and $g(\mathbf{x};\boldsymbol{\mu})$ are equal at the interpolation points $\{\mathbf{z}_m\}_{m=1}^M$.

Further note that we can express $g_M(\mathbf{x};\boldsymbol{\mu})$ in terms of the cardinal functions (Lagrange interpolation functions) as

$$g_M(\mathbf{x};\boldsymbol{\mu}) = \sum_{m=1}^{M} g(\mathbf{z}_m;\boldsymbol{\mu})\psi_m(\mathbf{x}) \tag{3}$$

Here, the cardinal functions $\{\psi_m\}_{m=1}^M$ are defined by $\psi_j(\mathbf{z}_i) = \delta_{ij}$ and hence are given by

$$\phi_i(\mathbf{x}) = \sum_{j=1}^{M} \phi_i(\mathbf{z}_j)\psi_j(\mathbf{x}), \quad 1 \leqslant i \leqslant M \tag{4}$$

We point out that $\{\psi_m\}_{m=1}^M$ depends on both $\{\phi_m\}_{m=1}^M$ and $\{\mathbf{z}_m\}_{m=1}^M$.

The approximation quality depends critically on both the basis set $\{\phi_m\}_{m=1}^M$ and the point set $\{\mathbf{z}_m\}_{m=1}^M$. In this paper, we use the POD method [36–38] to compute $\{\phi_m\}_{m=1}^M$ from a suitably fine set of snapshots

$$\mathscr{S}_K^g \equiv \{\xi_k^g(\mathbf{x}) = g(\mathbf{x};\boldsymbol{\mu}_k^g),\ \boldsymbol{\mu}_k^g \in S_K^g, 1 \leqslant k \leqslant K\} \tag{5}$$

where $S_K^g = \{\boldsymbol{\mu}_1^g, \ldots, \boldsymbol{\mu}_K^g\}$ is a selected parameter sample set. Details of the POD procedure are given in Appendix A for reference. Once we have the basis set $\{\phi_m\}_{m=1}^M$, we can determine the point set $\{\mathbf{z}_m\}_{m=1}^M$ as follows.

2.2. Interpolation points

We first introduce the best approximations of the elements in the snapshot set as

$$g_M^*(\cdot;\boldsymbol{\mu}_k^g) = \arg \min_{w_M \in \Phi_M} \|g(\cdot;\boldsymbol{\mu}_k^g) - w_M\|, \quad 1 \leqslant k \leqslant K \tag{6}$$

DOI: 10.1002/nme

It is easily derived from the orthonormality of the ϕ_m that

$$g_M^*(\mathbf{x};\boldsymbol{\mu}_k^g)=\sum_{m=1}^{M}\alpha_{Mm}(\boldsymbol{\mu}_k^g)\phi_m(\mathbf{x}),\quad 1\leqslant k\leqslant K \tag{7}$$

where the coefficients are given by

$$\alpha_{Mm}(\boldsymbol{\mu}_k^g)=(\phi_m, g(\cdot;\boldsymbol{\mu}_k^g)),\quad m=1,\ldots,M,\ \ 1\leqslant k\leqslant K \tag{8}$$

We now define our interpolation points $\{\mathbf{z}_m\}_{m=1}^M$ as a minimizer of the following minimization problem:

$$\min_{\mathbf{x}_1\in\Omega,\ldots,\mathbf{x}_M\in\Omega}\sum_{k=1}^{K}\left\|g_M^*(\cdot;\boldsymbol{\mu}_k^g)-\sum_{m=1}^{M}\beta_{Mm}(\mathbf{x}_1,\ldots,\mathbf{x}_M;\boldsymbol{\mu}_k^g)\phi_m\right\|^2$$

$$\sum_{n=1}^{M}\phi_n(\mathbf{x}_m)\beta_{Mn}(\mathbf{x}_1,\ldots,\mathbf{x}_M;\boldsymbol{\mu}_k^g)=g(\mathbf{x}_m;\boldsymbol{\mu}_k^g),\quad 1\leqslant m\leqslant M,\ \ 1\leqslant k\leqslant K \tag{9}$$

Substituting (8) into (9) and invoking orthonormality of $\{\phi_m\}_{m=1}^M$, we obtain

$$\min_{\mathbf{x}_1\in\Omega,\ldots,\mathbf{x}_M\in\Omega}\sum_{k=1}^{K}\sum_{m=1}^{M}(\alpha_{Mm}(\boldsymbol{\mu}_k^g)-\beta_{Mm}(\mathbf{x}_1,\ldots,\mathbf{x}_M;\boldsymbol{\mu}_k^g))^2$$

$$\sum_{n=1}^{M}\phi_n(\mathbf{x}_m)\beta_{Mn}(\mathbf{x}_1,\ldots,\mathbf{x}_M;\boldsymbol{\mu}_k^g)=g(\mathbf{x}_m;\boldsymbol{\mu}_k^g),\quad 1\leqslant m\leqslant M,\ \ 1\leqslant k\leqslant K \tag{10}$$

In words, $\{\mathbf{z}_m\}_{m=1}^M$ is determined so as to minimize the average error between the interpolants $g_M(\cdot;\boldsymbol{\mu}_k^g)$ and the best approximations $g_M^*(\cdot;\boldsymbol{\mu}_k^g)$. For this reason, the points $\{\mathbf{z}_m\}_{m=1}^M$ shall be referred as 'best points'.[§]

2.3. Solution procedure

We now find a solution to the least-squares minimization problem (10) using the Levenberg–Marquardt (LM) algorithm [39]. Let $\mathbf{s}=(\mathbf{x}_1,\ldots,\mathbf{x}_M)$; we write the objective in (10) as

$$F(\mathbf{s})=\frac{1}{2}\sum_{q=1}^{Q}r_q^2(\mathbf{s}) \tag{11}$$

where $r_q(\mathbf{s}), 1\leqslant q\leqslant Q=KM$, are given by

$$r_q(\mathbf{s})=\alpha_{Nm}(\boldsymbol{\mu}_k^g)-\beta_{Nm}(\mathbf{s};\boldsymbol{\mu}_k^g),\quad 1\leqslant k\leqslant K,\ \ 1\leqslant m\leqslant M \tag{12}$$

[§] Although BPIM and EIM address the same problem, the two approaches differ significantly in several ways. The EIM [35] constructs the basis set and the interpolation point set through an inductive interpolation procedure, which chooses the Lagrange basis functions at the parameter points where the L^∞ error attains maximum value and the interpolation points at the spatial coordinates where the error function attains its largest absolute magnitude. Further details of the BPIM and its relation to the EIM can be found in [34].

DOI: 10.1002/nme

The gradient and Hessian of the objective function $F(\mathbf{s})$ can thus be computed as

$$\nabla F(\mathbf{s}) = \sum_{q=1}^{Q} r_q(\mathbf{s}) \nabla r_q(\mathbf{s}) = (\mathbf{J}(\mathbf{s}))^{\mathrm{T}} \mathbf{r}(\mathbf{s}) \tag{13}$$

$$\nabla^2 F(\mathbf{s}) = (\mathbf{J}(\mathbf{s}))^{\mathrm{T}} \mathbf{J}(\mathbf{s}) + \sum_{q=1}^{Q} r_q(\mathbf{s}) \nabla^2 r_q(\mathbf{s}) \tag{14}$$

where for $1 \leqslant q \leqslant Q$, $1 \leqslant m \leqslant M$,

$$J_{qm}(\mathbf{s}) = \frac{\partial r_q(\mathbf{s})}{\partial \mathbf{x}_m} = \frac{\partial \beta_{Mm}(\mathbf{s}; \boldsymbol{\mu}_k^g)}{\partial \mathbf{x}_m} \tag{15}$$

Hence, when the residuals $r_q(\mathbf{s})$ are small, we may approximately compute the Hessian in terms of only the Jacobian matrix $\mathbf{J}(\mathbf{s})$ as

$$\nabla^2 F(\mathbf{s}) \approx (\mathbf{J}(\mathbf{s}))^{\mathrm{T}} \mathbf{J}(\mathbf{s}) \tag{16}$$

The Jacobian matrix $\mathbf{J}(\mathbf{s})$ is computed by differentiating both sides of constraint (10) with respect to $\mathbf{s}$. Finally, the interpolation points $\mathbf{z}_m$, $1 \leqslant m \leqslant M$, are determined by solving problem (10). For the examples presented in this paper, the optimal solution is typically reached in less than 15 iterations of the LM algorithm.

3. NONAFFINE LINEAR ELLIPTIC EQUATIONS

In this section, we develop the reduced-order approximation of linear elliptic equations with nonaffine parameter dependence. The basic idea is to replace the nonaffine terms with the coefficient-function expansions constructed using the BPIM. A model problem involving geometric variation is provided to demonstrate the application of our ROA to shape optimization problems. Note that we can also develop the *a posteriori* error estimation for the reduced-order model discussed in this section. However, since such extension goes beyond the content of this paper, we refer the reader to [32] for a detailed discussion.

3.1. Abstract problem formulation

We consider a suitably regular domain $\Omega \subset \mathbb{R}^d$, $d = 2$, with spatial coordinate $\mathbf{x}$. We define the Hilbert spaces $X \equiv H_0^1(\Omega)$—or, more generally, $H_0^1(\Omega) \subset X \subset H^1(\Omega)$—where $H^1(\Omega) = \{v | v \in L^2(\Omega), \nabla v \in (L^2(\Omega))^d\}$ and $H_0^1(\Omega) = \{v | v \in H^1(\Omega), v|_{\partial\Omega} = 0\}$. We also recall the parameter domain $\mathscr{D} \subset \mathbb{R}^P$ and parameter vector $\boldsymbol{\mu} = (\mu^1, \ldots, \mu^P)$ introduced in the previous section.

The abstract formulation for a $\boldsymbol{\mu}$-parametrized linear elliptic PDE can be stated as follows: given any $\boldsymbol{\mu} \in \mathscr{D} \subset \mathbb{R}^P$, we find

$$s(\boldsymbol{\mu}) = \ell^O(u(\boldsymbol{\mu}); \boldsymbol{\mu}) \tag{17}$$

where $u(\boldsymbol{\mu}) \in X(\Omega)$ is the solution to

$$a(u(\boldsymbol{\mu}), v; \boldsymbol{\mu}) = \ell(v; \boldsymbol{\mu}) \quad \forall v \in X(\Omega) \tag{18}$$

DOI: 10.1002/nme

Here, $\ell^O(\cdot;\boldsymbol{\mu}), \ell(\cdot;\boldsymbol{\mu})$ are continuous functionals, and $a(\cdot,\cdot;\boldsymbol{\mu})$ is a continuous bounded bilinear form. In addition, we make certain assumptions on the parametric dependence of a, ℓ, and ℓ^O. In particular, we suppose that, for some finite Q, a may be expressed as

$$a(w,v;\boldsymbol{\mu}) = \sum_{q=1}^{Q} a^q(w,v,g^q(\cdot;\boldsymbol{\mu})) \tag{19}$$

and ℓ and ℓ^0 are given by

$$\ell(v;\boldsymbol{\mu}) = b(v,h(\cdot;\boldsymbol{\mu})) \tag{20}$$

$$\ell^O(v;\boldsymbol{\mu}) = c(v,h^O(\cdot;\boldsymbol{\mu})) \tag{21}$$

Here $a^q: X\times X\times L^\infty(\Omega)\to\mathbb{R}, 1\leqslant q\leqslant Q$, are continuous $\boldsymbol{\mu}$-dependent *trilinear* forms; $b: X\times L^\infty(\Omega)\to\mathbb{R}$ and $c: X\times L^\infty(\Omega)\to\mathbb{R}$ are continuous $\boldsymbol{\mu}$-dependent *bilinear* forms; and $h(\mathbf{x};\boldsymbol{\mu})$, $h^O(\mathbf{x};\boldsymbol{\mu})$, $g^q(\mathbf{x};\boldsymbol{\mu})\in L^\infty(\Omega)\cap C^0(\Omega): \Omega\times\mathscr{D}\to\mathbb{R}, 1\leqslant q\leqslant Q$, are known functions. We note that a defined above is *nonaffine* in the parameter because the operators a^q depend on $g^q(\mathbf{x};\boldsymbol{\mu})$—general functions of spatial coordinate $\mathbf{x}$ and parameter vector $\boldsymbol{\mu}$.

In practice, the exact solution $u(\boldsymbol{\mu})$ is often not available; we thus replace $u(\boldsymbol{\mu})$ with a 'truth' approximation, $u_h(\boldsymbol{\mu})$, which resides in (say) a suitably fine piecewise-linear FE approximation space $X_h\subset X$ of *very* large dimension $\mathscr{N}$. The FE discretization of (17)–(18) is thus: given any $\boldsymbol{\mu}\in\mathscr{D}$, we evaluate

$$s_h(\boldsymbol{\mu}) = \ell^O(u_h(\boldsymbol{\mu});\boldsymbol{\mu}) \tag{22}$$

where $u_h(\boldsymbol{\mu})\in X_h$ satisfies

$$a(u_h(\boldsymbol{\mu}),v;\boldsymbol{\mu}) = \ell(v;\boldsymbol{\mu}) \quad \forall v\in X_h \tag{23}$$

We shall assume that the discretization is sufficiently rich such that $u_h(\boldsymbol{\mu})$ and $u(\boldsymbol{\mu})$ and hence $s_h(\boldsymbol{\mu})$ and $s(\boldsymbol{\mu})$ are indistinguishable at the accuracy level of interest. Unfortunately, the computational cost associated with this FE approximation will depend on some power of $\mathscr{N}$. As a result, the evaluation $\boldsymbol{\mu}\to s_h(\boldsymbol{\mu})$ is simply too costly in the many-query and real-time contexts often of interest in engineering.

In what follows, we develop a reduced-order approximation of problem (17)–(18) for the rapid prediction of the output of interest. The reduced-order approximation shall be built upon the FE approximation, and the reduced-order error will thus be evaluated with respect to $s_h(\boldsymbol{\mu})$.

3.2. Reduced-order approximation

We follow the formulation outlined in [33] to derive the discrete reduced system. To begin, we introduce a parameter sample $S_K^u = \{\boldsymbol{\mu}_1^u\in\mathscr{D},\ldots,\boldsymbol{\mu}_K^u\in\mathscr{D}\}$ and an associated set of snapshots $\mathscr{S}_K^u \equiv \{\xi_k^u = u_h(\boldsymbol{\mu}_k^u), 1\leqslant k\leqslant K\}$, where $u_h(\boldsymbol{\mu}_k^u)$ is a solution to (23) for $\boldsymbol{\mu}=\boldsymbol{\mu}_k^u$. Upon the snapshot set $\mathscr{S}_K^u$, we apply the POD procedure (see in Appendix A) to construct N ($\leqslant K$) basis functions $\{\zeta_n\}_{n=1}^N$ and define an associated approximation space $W_N = \text{span}\{\zeta_1,\ldots,\zeta_N\}$. (We note that the greedy sampling algorithm based on the L^∞ norm is used in [33] to compute the Lagrange reduced-basis spaces. In [40], it has been numerically verified that the POD reduced basis yields smaller error in the L^2 norm, but larger error in the L^∞ norm with a very small difference. This discrepancy is

DOI: 10.1002/nme

likely due to the fact that the POD provides basis functions optimal in the L^2 norm, whereas the greedy approach provides optimal basis functions in the stronger L^∞ norm.)

Were we to follow the standard Galerkin ROA, we would obtain an approximation $u_N^{\mathrm{SG}}(\boldsymbol{\mu}) \in W_N$ that satisfies

$$\sum_{q=1}^{Q} a^q(u_N^{\mathrm{SG}}(\boldsymbol{\mu}), v, g^q(\cdot;\boldsymbol{\mu})) = b(v, h(\cdot;\boldsymbol{\mu})) \quad \forall v \in W_N \tag{24}$$

Expressing $u_N^{\mathrm{SG}}(\boldsymbol{\mu}) = \sum_{j=1}^{N} u_{Nj}^{\mathrm{SG}}(\boldsymbol{\mu})\zeta_j$ and choosing $v = \zeta_i$, $1 \leqslant i \leqslant N$, we arrive at

$$\mathbf{A}_N^{\mathrm{SG}}(\boldsymbol{\mu})\mathbf{u}_N^{\mathrm{SG}}(\boldsymbol{\mu}) = \mathbf{L}_N^{\mathrm{SG}}(\boldsymbol{\mu}) \tag{25}$$

where $\mathbf{A}_N^{\mathrm{SG}}(\boldsymbol{\mu}) \in \mathbb{R}^{N\times N}$ and $\mathbf{L}_N^{\mathrm{SG}}(\boldsymbol{\mu}) \in \mathbb{R}^N$ are given by

$$\begin{aligned} A_{Nij}^{\mathrm{SG}}(\boldsymbol{\mu}) &= \sum_{q=1}^{Q} a^q(\zeta_j, \zeta_i, g^q(\cdot;\boldsymbol{\mu})), \quad 1 \leqslant i, \quad j \leqslant N \\ L_{Ni}^{\mathrm{SG}}(\boldsymbol{\mu}) &= b(\zeta_i, h(\cdot;\boldsymbol{\mu})), \quad 1 \leqslant i \leqslant N \end{aligned} \tag{26}$$

The output of interest would then be evaluated as

$$s_N^{\mathrm{SG}}(\boldsymbol{\mu}) = \sum_{j=1}^{N} u_{Nj}^{\mathrm{SG}}(\boldsymbol{\mu}) c(\zeta_j, h^O(\cdot;\boldsymbol{\mu})) \tag{27}$$

We note that since $N \ll \mathcal{N}$ the (full) linear system (25) of size $N \times N$ is very small compared with the (sparse) $\mathcal{N} \times \mathcal{N}$ linear system associated with the FE discretization (23). The standard Galerkin ROA can thus effect a significant reduction in the degrees of freedom. Unfortunately, for any given parameter $\boldsymbol{\mu}$, the cost of assembling the stiffness matrix $\mathbf{A}_N^{\mathrm{SG}}(\boldsymbol{\mu})$ depends on $\mathcal{N}$ because of the presence of the nonaffine functions $g^q(\mathbf{x}, \boldsymbol{\mu})$, $1 \leqslant q \leqslant Q$, in the trilinear forms a^q, $1 \leqslant q \leqslant Q$. Similarly, the cost of assembling $L_{Ni}^{\mathrm{SG}}(\boldsymbol{\mu})$ and evaluating the output $s_N^{\mathrm{SG}}(\boldsymbol{\mu})$ also scales with $\mathcal{N}$. Consequently, although having much less degrees of freedom, the standard Galerkin reduced-order model may not offer significant computational savings relative to the FE approximation using advanced iterative methods.

To obtain an efficient reduced-order approximation, we first employ the BPIM to construct the point sets $\{\mathbf{z}_m^{g^1}\}_{m=1}^{M^{g^1}}, \ldots, \{\mathbf{z}_m^{g^Q}\}_{m=1}^{M^{g^Q}}$, $\{\mathbf{z}_m^h\}_{m=1}^{M^h}$, $\{\mathbf{z}_m^{h^O}\}_{m=1}^{M^{h^O}}$ and basis sets $\{\psi_m^{g^1}\}_{m=1}^{M^{g^1}}, \ldots, \{\psi_m^{g^Q}\}_{m=1}^{M^{g^Q}}$, $\{\psi_m^h\}_{m=1}^{M^h}$, $\{\psi_m^{h^O}\}_{m=1}^{M^{h^O}}$ for $g^q(\mathbf{x};\boldsymbol{\mu})$, $1 \leqslant q \leqslant Q$, $h(\mathbf{x};\boldsymbol{\mu})$, and $h^O(\mathbf{x};\boldsymbol{\mu})$ respectively, as described in Section 2. For notational simplification, we presume that $M^{g^1} = \cdots = M^{g^Q} = M^h = M^{h^O} = M$. We next replace $g^q(\mathbf{x}, \boldsymbol{\mu})$, $1 \leqslant q \leqslant Q$, $h(\mathbf{x};\boldsymbol{\mu})$, and $h^O(\mathbf{x};\boldsymbol{\mu})$ with our coefficient-function expansions $g_M^q(\mathbf{x}, \boldsymbol{\mu})$, $1 \leqslant q \leqslant Q$, $h_M(\mathbf{x};\boldsymbol{\mu})$, and $h_M^O(\mathbf{x};\boldsymbol{\mu})$, respectively. Our reduced-order approximation is thus: for any given $\boldsymbol{\mu} \in \mathcal{D}$, we evaluate

$$s_{N,M}(\boldsymbol{\mu}) = c(u_{N,M}(\boldsymbol{\mu}), h_M^O(\cdot;\boldsymbol{\mu})) \tag{28}$$

where $u_{N,M}(\boldsymbol{\mu}) \in W_N$ satisfies

$$\sum_{q=1}^{Q} a^q(u_{N,M}(\boldsymbol{\mu}), v, g_M^q(\cdot;\boldsymbol{\mu})) = b(v, h_M(\cdot;\boldsymbol{\mu})) \quad \forall v \in W_N \tag{29}$$

DOI: 10.1002/nme

In order to derive the discrete equations for this reduced-order approximation, we write

$$\begin{aligned} u_{N,M}(\boldsymbol{\mu}) &= \sum_{n=1}^{N} u_{N,Mj}(\boldsymbol{\mu})\zeta_j \\ g_M^q(\cdot,\boldsymbol{\mu}) &= \sum_{m=1}^{M} g(\mathbf{z}_m^{g^q};\boldsymbol{\mu})\psi_m^{g^q}, \quad 1\leqslant q\leqslant Q \\ h_M(\cdot,\boldsymbol{\mu}) &= \sum_{m=1}^{M} h(\mathbf{z}_m^h;\boldsymbol{\mu})\psi_m^h \\ h_M^O(\cdot,\boldsymbol{\mu}) &= \sum_{m=1}^{M} h^O(\mathbf{z}_m^{h^O};\boldsymbol{\mu})\psi_m^{h^O} \end{aligned} \tag{30}$$

Substituting these representations into (29) and (28) and choosing $v=\zeta_i, 1\leqslant i\leqslant N$, we obtain

$$\sum_{j=1}^{N}\left(\sum_{q=1}^{Q}\sum_{m=1}^{M} g^q(\mathbf{z}_m^{g^q};\boldsymbol{\mu})a^q(\zeta_j,\zeta_i,\psi_m^{g^q})\right)u_{N,Mj}(\boldsymbol{\mu}) = \sum_{m=1}^{M} h(\mathbf{z}_m^h;\boldsymbol{\mu})b(\zeta_i,\psi_m^h), \quad 1\leqslant i\leqslant N \tag{31}$$

and

$$s_{N,M}(\boldsymbol{\mu}) = \sum_{m=1}^{M}\sum_{j=1}^{N} h^O(\mathbf{z}_m^{h^O};\boldsymbol{\mu})u_{N,Mj}(\boldsymbol{\mu})c(\zeta_j,\psi_m^{h^O}) \tag{32}$$

Equivalently, we arrive at the linear system of N equations for the coefficient vector $\mathbf{u}_{N,M}(\boldsymbol{\mu})$

$$\mathbf{A}_N(\boldsymbol{\mu})\mathbf{u}_{N,M}(\boldsymbol{\mu}) = \mathbf{L}_N(\boldsymbol{\mu}) \tag{33}$$

and then calculate

$$s_{N,M}(\boldsymbol{\mu}) = (\mathbf{L}_N^O(\boldsymbol{\mu}))^{\mathrm{T}}\mathbf{u}_{N,M}(\boldsymbol{\mu}) \tag{34}$$

where

$$\begin{aligned} \mathbf{A}_N(\boldsymbol{\mu}) &= \sum_{q=1}^{Q}\sum_{m=1}^{M} g^q(\mathbf{z}_m^{g^q};\boldsymbol{\mu})\mathbf{A}_N^{qm} \\ \mathbf{L}_N(\boldsymbol{\mu}) &= \sum_{m=1}^{M} h(\mathbf{z}_m^h,\boldsymbol{\mu})\mathbf{L}_N^m \\ \mathbf{L}_N^O(\boldsymbol{\mu}) &= \sum_{m=1}^{M} h^O(\mathbf{z}_m^{h^O},\boldsymbol{\mu})\mathbf{L}_N^{Om} \end{aligned} \tag{35}$$

Here $\mathbf{A}_N^{qm}\in\mathbb{R}^{N\times N}, 1\leqslant q\leqslant Q, 1\leqslant m\leqslant M$, and $\mathbf{L}_N^m, \mathbf{L}_N^{Om}\in\mathbb{R}^N, 1\leqslant m\leqslant M$, are given by

$$\begin{aligned} A_{Nij}^{qm} &= a^q(\zeta_i,\zeta_j,\psi_m^{g^q}), \quad 1\leqslant i,\ \ j\leqslant N \\ L_{Ni}^m &= b(\zeta_i,\psi_m^h), \quad 1\leqslant i\leqslant N \\ L_{Ni}^{Om} &= c(\zeta_i,\psi_m^{h^O}), \quad 1\leqslant i\leqslant N \end{aligned} \tag{36}$$

DOI: 10.1002/nme

Offline Stage

1. Compute $\{\psi_m^h\}_{m=1}^{M_{\max}}, \{\boldsymbol{z}_m^h\}_{m=1}^{M_{\max}}, \{\psi_m^{h^O}\}_{m=1}^{M_{\max}}, \{\boldsymbol{z}_m^{h^O}\}_{m=1}^{M_{\max}}$, $\{\psi_m^{g^q}\}_{m=1}^{M_{\max}}, \{\boldsymbol{z}_m^{g^q}\}_{m=1}^{M_{\max}}, 1 \le q \le Q,$ following Section 2;
2. Compute the snapshot set $\mathcal{S}_K^u \equiv \{\xi_k^u = u_h(\boldsymbol{\mu}_k^u), 1 \le k \le K\}$;
3. Compute the reduced basis set$\{\zeta_n\}_{n=1}^{N_{\max}}$ using the POD method;
4. Form and store: $\boldsymbol{A}_N^{q\,m}, 1 \le q \le Q, 1 \le m \le M_{\max}$, $\boldsymbol{L}_N^m, \boldsymbol{L}_N^{O\,m}, 1 \le m \le M_{\max},$ for $N = N_{\max}$.

Figure 1. Offline construction algorithm.

Online Stage

1. Choose $N(\le N_{\max})$ and $M(\le M_{\max})$;
2. Assemble $\boldsymbol{A}_N(\boldsymbol{\mu})$ from (35) at cost $O(QMN^2)$;
3. Assemble $\boldsymbol{L}_N(\boldsymbol{\mu}), \boldsymbol{L}_N^O(\boldsymbol{\mu})$ from (35) at cost $O(MN)$;
4. Solve $\boldsymbol{A}_N(\boldsymbol{\mu})\boldsymbol{u}_{N,M}(\boldsymbol{\mu}) = \boldsymbol{L}_N(\boldsymbol{\mu})$ at cost $O(N^3)$;
5. Calculate $s_{N,M}(\boldsymbol{\mu}) = (\boldsymbol{L}_N^O(\boldsymbol{\mu}))^T \boldsymbol{u}_{N,M}(\boldsymbol{\mu})$ at cost $O(N)$.

Figure 2. Online evaluation algorithm.

The matrices and vectors in the above equation do *not* depend on the parameter vector $\boldsymbol{\mu}$ and can therefore be pre-computed offline. The remaining operations required to form (33)–(35) can be performed online.

We summarize the computational process in Figure 1 for the offline stage and in Figure 2 for the online stage. The offline construction of our reduced-order model is of course computationally intensive, but it is done only *one time*. However, in the online stage, we can compute $s_{N,M}(\boldsymbol{\mu})$ with a total cost of $O(N^3 + QMN^2)$ operations for any given new parameter vector $\boldsymbol{\mu}$.

Hence, as required in the many-query or real-time contexts, the online complexity is *independent* of $\mathcal{N}$—the dimension of the FE approximation space. Since $N, M, Q \ll \mathcal{N}$, we expect significant computational savings in the online stage relative to the FE approximation (23) and relative to the standard Galerkin reduced-order model built upon (24).

3.3. Example 1: a problem of geometric variation

We consider the Poisson equation with homogeneous Dirichlet condition solved on a parametrized domain $\hat{\Omega}(\boldsymbol{\mu})$,

$$-\nabla^2 \Psi = 1 \quad \text{in } \hat{\Omega}(\boldsymbol{\mu})$$

$$\Psi = 0 \quad \text{on } \partial\hat{\Omega}$$

The parameter vector is given by $\boldsymbol{\mu} = (\mu^1, \mu^2) \equiv (R, \kappa) \in \mathscr{D} \equiv [1, 10] \times [1, 10]$, where R and κ control the size and shape of the domain as shown in Figure 3; the inner and outer boundaries of the cross-section are described by the closed curves $|\hat{x}^1|^\kappa + |\hat{x}^2|^\kappa = 1$ and $|\hat{x}^1|^\kappa + |\hat{x}^2|^\kappa = (R+1)^\kappa$, respectively. Finally, the output of interest is the integral of Ψ over $\hat{\Omega}$. Although simple enough, this example serves to demonstrate the usefulness of our ROA for shape optimization problems.

DOI: 10.1002/nme

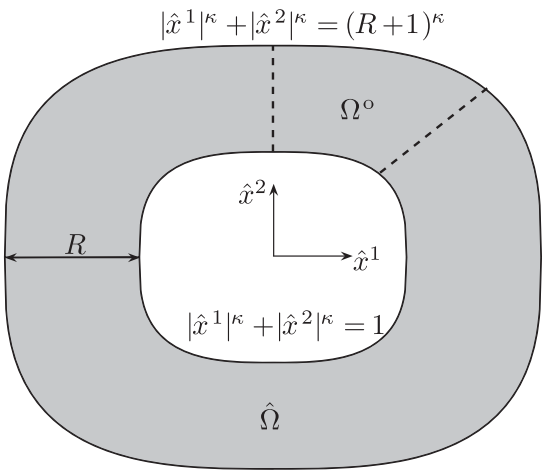

Figure 3. The cross-section $\hat{\Omega}$ varies with geometric parameters R and κ. The 'cut' domain Ω^o is formed by the dashed lines and the boundary of the cross-section.

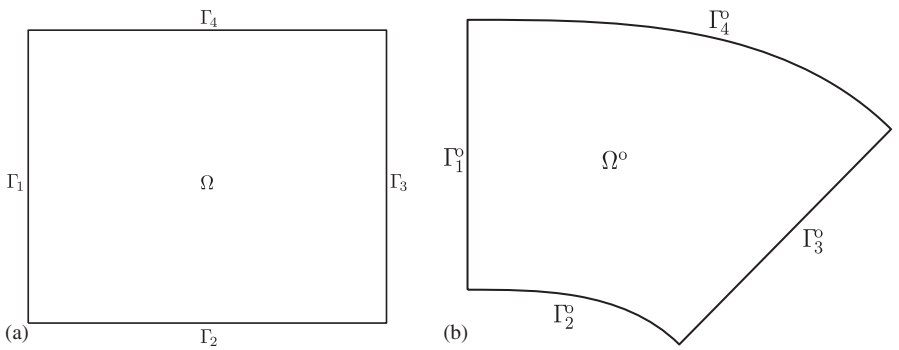

Figure 4. The computational domains: (a) reference domain Ω and (b) original cut domain Ω^o.

Owing to geometric symmetry, the problem can be reformulated as $-\nabla^2 u^o = 1$ in $\Omega^o(\boldsymbol{\mu})$ with homogeneous Neumann condition on $\Gamma_N^o \equiv \Gamma_1^o \cup \Gamma_3^o$ and homogeneous Dirichlet condition on $\Gamma_D^o \equiv \Gamma_2^o \cup \Gamma_4^o$; here $\Omega^o(\boldsymbol{\mu})$ is a 'cut' domain as shown in Figure 4(b). Hence, $u^o(\boldsymbol{\mu}) \in X^o$, $X^o \equiv \{v \in H^1(\Omega^o(\boldsymbol{\mu}))|v|_{\Gamma_D^o} = 0\}$, is the solution to

$$\int_{\Omega^o(\boldsymbol{\mu})} \nabla u^o(\boldsymbol{\mu}) \cdot \nabla v \, d\Omega^o = \int_{\Omega^o(\boldsymbol{\mu})} v \, d\Omega^o \quad \forall v \in H^1(\Omega^o(\boldsymbol{\mu})) \tag{37}$$

The output of interest is then $s^o(\boldsymbol{\mu}) = 8 \int_{\Omega^o(\boldsymbol{\mu})} u^o(\boldsymbol{\mu}) \, d\Omega^o$.

We treat geometric variation in an indirect way by transforming Equation (37) on $\Omega^o(\boldsymbol{\mu})$ to a new equation on a 'fixed' reference domain $\Omega \equiv]0,1[\times]0,1[$ shown in Figure 4(a). To do this,

DOI: 10.1002/nme

we establish a one-to-one mapping between Ω and $\Omega^{\mathrm{o}}(\boldsymbol{\mu})$. The geometric mapping $\mathscr{F}$ from $\mathbf{x}=(x^1,x^2)\in\Omega$ to $\mathbf{x}_{\mathrm{o}}=(x^1_{\mathrm{o}},x^2_{\mathrm{o}})\in\Omega^{\mathrm{o}}(\boldsymbol{\mu})$ is given by

$$x^1_{\mathrm{o}}=\frac{(x^1)((x^2)R+1)}{(1+(x^1)^\kappa)^{1/\kappa}},\quad x^2_{\mathrm{o}}=\frac{(x^2)R+1}{(1+(x^1)^\kappa)^{1/\kappa}} \tag{38}$$

Our exact solution on the original domain, $u^{\mathrm{o}}(\boldsymbol{\mu})$, can then be expressed in terms of the solution on the mapped domain, $u(\boldsymbol{\mu})$, as $u^{\mathrm{o}}(\mathbf{x}_{\mathrm{o}};\boldsymbol{\mu})=u(\mathscr{F}^{-1}(\mathbf{x}_{\mathrm{o}});\boldsymbol{\mu})$. The solution on the mapped domain satisfies a weak formulation of the form (18) in which the trilinear forms, bilinear forms, and nonaffine functions are given by

$$\begin{aligned}
a^1(w,v,g^1(\mathbf{x};\boldsymbol{\mu}))&=\int_\Omega g^1(\mathbf{x};\boldsymbol{\mu})\frac{\partial w}{\partial x^1}\frac{\partial v}{\partial x^1}\,\mathrm{d}\Omega\\
a^2(w,v,g^2(\mathbf{x};\boldsymbol{\mu}))&=\int_\Omega g^2(\mathbf{x};\boldsymbol{\mu})\frac{\partial w}{\partial x^2}\frac{\partial v}{\partial x^2}\,\mathrm{d}\Omega\\
a^3(w,v,g^3(\mathbf{x};\boldsymbol{\mu}))&=\int_\Omega g^3(\mathbf{x};\boldsymbol{\mu})\left(\frac{\partial w}{\partial x^1}\frac{\partial v}{\partial x^2}+\frac{\partial w}{\partial x^2}\frac{\partial v}{\partial x^1}\right)\mathrm{d}\Omega\\
b(v,h(\mathbf{x};\boldsymbol{\mu}))&=\int_\Omega h(\mathbf{x};\boldsymbol{\mu})v\,\mathrm{d}\Omega\\
c(v,h^O(\mathbf{x};\boldsymbol{\mu}))&=\int_\Omega h^O(\mathbf{x};\boldsymbol{\mu})v\,\mathrm{d}\Omega\\
g^1(\mathbf{x};\boldsymbol{\mu})&=\frac{R(1+(x^1)^2)}{R(x^2)+1}\\
g^2(\mathbf{x};\boldsymbol{\mu})&=\frac{R(x^1)^{2\kappa-2}(x^2)+R(x^2)+(x^1)^{2\kappa-2}+1}{R((x^1)^\kappa+1)^2}\\
g^3(\mathbf{x};\boldsymbol{\mu})&=\frac{(x^1)^{\kappa-1}-(x^1)}{(x^1)^\kappa+1}\\
h(\mathbf{x};\boldsymbol{\mu})&=R(1+(x^1)^\kappa)^{-2/\kappa}(R(x^2)+1)\\
h^O(\mathbf{x};\boldsymbol{\mu})&=R(1+(x^1)^\kappa)^{-2/\kappa}(R(x^2)+1)
\end{aligned}$$

Here $X\equiv\{v\in H^1(\Omega)|v|_{\Gamma^1\cup\Gamma_3}=0\}$. The output is evaluated as $s(\boldsymbol{\mu})=\ell^O(u(\boldsymbol{\mu});\boldsymbol{\mu})$; note for this example that $\ell^O=\ell$ and $h^O(\mathbf{x};\boldsymbol{\mu})=h(\mathbf{x};\boldsymbol{\mu})$.

The computational domain is discretized uniformly into piecewise-linear resulting in an FE approximation space $X_h\in X$ of dimension $\mathcal{N}=10\,000$. We present in Figure 5 the computed solutions for different parameter values. As regards the offline construction, we choose for S^g_K a regular grid of 15×15 points over $\mathscr{D}$ and construct $\{\psi^h_m\}_{m=1}^{M_{\max}}$, $\{\mathbf{z}^h_m\}_{m=1}^{M_{\max}}$, and $\{\psi^{g^q}_m\}_{m=1}^{M_{\max}}$, $\{\mathbf{z}^{g^q}_m\}_{m=1}^{M_{\max}}$, $1\leqslant q\leqslant Q$, for $M_{\max}=12$; we then take $\mathscr{S}^u_K=\mathscr{S}^g_K$ and compute the associated snapshot set $\mathscr{S}^u_K$ upon which the basis set $\{\zeta_n\}_{n=1}^{N_{\max}}$ is constructed for $N_{\max}=30$. We plot in Figure 6 the point sets corresponding to $g^2(\mathbf{x};\boldsymbol{\mu})$ and $h(\mathbf{x};\boldsymbol{\mu})$ for $M=M_{\max}$. We see that most interpolation points are

DOI: 10.1002/nme

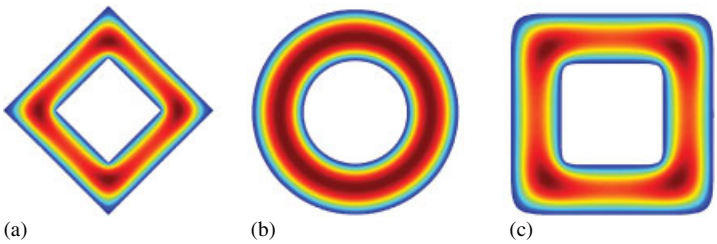

Figure 5. The FE solutions on the physical domain $\hat{\Omega}$ for: (a) $\boldsymbol{\mu}=(1,1)$, (b) $\boldsymbol{\mu}=(1,2)$, and (c) $\boldsymbol{\mu}=(1,10)$. Note how the geometry and solution change as μ^2 increases.

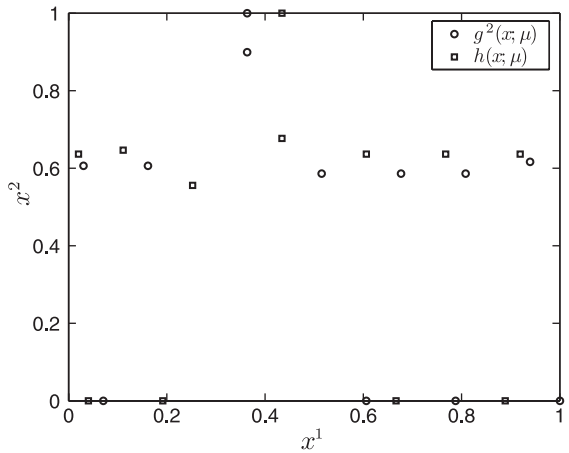

Figure 6. Distribution of the interpolation points on the reference domain Ω for $g^2(\mathbf{x};\boldsymbol{\mu})$ and $h(\mathbf{x};\boldsymbol{\mu})$.

distributed along the $x^2=0.6$ and $x^2=0$ lines. Since these functions are linear in x^2, only two points suffice to capture their behavior in the x^2 direction.

We now present results obtained with our ROA. For this purpose, we choose a test sample Ξ_{Test} as a regular 20×20 grid over $\mathscr{D}$ and define the average relative error norm as

$$\varepsilon^{u}_{\text{ave,rel}} = \text{mean}_{\boldsymbol{\mu}\in\Xi_{\text{Test}}} \frac{\|u_h(\boldsymbol{\mu}) - u_{N,M}(\boldsymbol{\mu})\|}{\|u_h(\boldsymbol{\mu})\|}$$

Figure 7 shows $\varepsilon^{u}_{\text{ave,rel}}$ as a function of N and M. We observe very rapid convergence of $u_{N,M}(\boldsymbol{\mu})$ to $u_h(\boldsymbol{\mu})$. The quality of the reduced-order approximation depends on N and M in a strongly coupled manner: for a fixed value of M the error initially decreases with increasing N and then levels off for N large enough; when the error does not improve with increasing N, increasing M tends to reduce the error. This behavior of the error is expected because the accuracy of our reduced-order approximation is limited by the coefficient-function approximation error which is decreased with increasing M. It basically suggests that optimal combinations of N and M are at the 'knees' of the error curves: for example, the combination $M=6$ and $N=12$ appears to be nearly optimal. Furthermore, we note that our coefficient-function approximation can lead to

DOI: 10.1002/nme

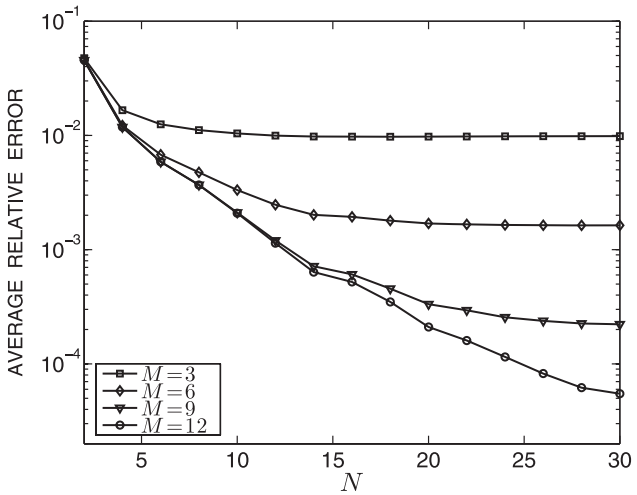

Figure 7. Average relative error norm $\varepsilon^{\mu}_{\text{ave,rel}}$ as a function of N and M for Example 1.

Table I. Numerical results for Example 1: maximum relative error in the output over the test sample Ξ_{Test} and normalized computational time as a function of N for the ROA (with $M=12$) and the SGA.

		ROA		SGA		FEA
N	M	Maximum relative error	Online time	Maximum relative error	Online time	Computational time
4	12	4.28E−02	2.34E−04	4.28E−02	1.94E−01	1
8	12	2.14E−02	5.75E−04	2.14E−02	1.95E−01	1
12	12	1.49E−02	9.41E−04	1.49E−02	1.97E−01	1
16	12	5.32E−03	1.23E−03	5.32E−03	2.01E−01	1
20	12	9.76E−04	1.59E−03	9.75E−04	2.04E−01	1

Note: The online times are normalized with respect to the computational time for $s_h(\boldsymbol{\mu})$.

a very accurate representation of the nonaffine terms and indeed, for this example, M can be chosen smaller than N without sacrificing the accuracy.

Finally, we compare the results obtained with the proposed ROA to those obtained with the standard Galerkin reduced-order approach (SGA). In Table I, we present the maximum relative error in the output over the test sample Ξ_{Test} and computational times as a function of N for both the ROA (with $M=12$) and the SGA. Here, the computational times are normalized with respect to the time required to compute the finite element approximation (FEA) output $s_h(\boldsymbol{\mu})$ shown in the last column of Table I. Since the standard Galerkin reduced-order model suffers from an $\mathcal{N}$*-dependent* cost of assembling the reduced stiffness matrix and load vector, its computational advantage relative to the FE approximation is very modest as observed in Table I. In contrast, our reduced-order approximation achieves significant computational savings relative to the FE approximation and yields a convergence rate which is very similar to that of the standard Galerkin reduced-order model. Of course, this comparison is meaningful only if we are in the real-time or many-query contexts—in which the offline cost can be amortized over *many* output predictions.

DOI: 10.1002/nme

4. NON-LINEAR ELLIPTIC EQUATIONS

In this section, we illustrate how the proposed approach can be applied to non-linear elliptic equations. The numerical difficulty is the presence of strong non-linearities in the differential operator; we shall simply treat non-linear terms as 'functions' and construct associated coefficient-function approximations. We shortly introduce the abstract formulation and describe the reduced-order approximation; we then discuss numerical results obtained for a model problem. Further discussion and in particular the development of *a posteriori* error estimation can be found in [41].

4.1. Abstract problem formulation

We consider the following problem: for any $\boldsymbol{\mu} \in \mathscr{D}$, find

$$s(\boldsymbol{\mu}) = \ell^O(u(\boldsymbol{\mu})) \tag{39}$$

where $u(\boldsymbol{\mu})$ satisfies the weak form of the $\boldsymbol{\mu}$-parametrized non-linear elliptic PDE

$$\Theta(\boldsymbol{\mu}) a_L(u(\boldsymbol{\mu}), v) + \int_\Omega g(u(\boldsymbol{\mu}); \boldsymbol{\mu}) v \, \mathrm{d}\Omega = \ell(v) \quad \forall v \in X \tag{40}$$

Here $g(w; \boldsymbol{\mu}): X \times \mathscr{D} \to \mathbb{R}$ is a general non-linear function of $w \in X$ and the parameter vector $\boldsymbol{\mu}$; $\Theta: \mathscr{D} \to \mathbb{R}$ is a $\boldsymbol{\mu}$-dependent function; $a_L: X \times X \to \mathbb{R}$ is a continuous bounded bilinear functional and $\ell^O, \ell: X \to \mathbb{R}$ are continuous bounded functionals. Here, for simplicity, we shall assume that a_L and ℓ^O, ℓ are independent of $\boldsymbol{\mu}$, although this limitation could be readily removed. Further, we assume that our abstract problem is well posed in the sense of Hadamard, meaning that its solution exists, is unique, and depends continuously on the data functional ℓ.

Next, we recall our FE approximation space $X_h (\subset X)$ of dimension $\mathscr{N}$. Our FE approximation is then: given $\boldsymbol{\mu} \in \mathscr{D}$, we find

$$s_h(\boldsymbol{\mu}) = \ell^O(u_h(\boldsymbol{\mu})) \tag{41}$$

where $u_h(\boldsymbol{\mu}) \in X_h$ is the solution to the discretized weak statement

$$\Theta(\boldsymbol{\mu}) a_L(u_h(\boldsymbol{\mu}), v) + \int_\Omega g(u_h(\boldsymbol{\mu}); \boldsymbol{\mu}) v \, \mathrm{d}\Omega = \ell(v) \quad \forall v \in X_h \tag{42}$$

We assume that $|s(\boldsymbol{\mu}) - s_h(\boldsymbol{\mu})|$ is suitably small and hence that $\mathscr{N}$ will typically be very large.

4.2. Reduced-order approximation

We first introduce a sample set $S_K^u = \{\boldsymbol{\mu}_1^u \in \mathscr{D}, \ldots, \boldsymbol{\mu}_K^u \in \mathscr{D}\}$, associated snapshot set $\mathscr{S}_K^u = \{\xi_k = u_h(\boldsymbol{\mu}_k^u), 1 \leqslant k \leqslant K\}$, and reduced-basis space $W_N = \mathrm{span}\{\zeta_n, 1 \leqslant n \leqslant N\}$; here $u_h(\boldsymbol{\mu}_k^u)$ is the solution of (42) at $\boldsymbol{\mu} = \boldsymbol{\mu}_k^u$ and ζ_n, $1 \leqslant n \leqslant N$, are the POD basis functions. The standard Galerkin reduced-order model [7, 15–18] is then obtained by a standard Galerkin projection: given $\boldsymbol{\mu} \in \mathscr{D}$, we evaluate

$$s_N^{\mathrm{SG}}(\boldsymbol{\mu}) = \ell^O(u_N^{\mathrm{SG}}(\boldsymbol{\mu})) \tag{43}$$

where $u_N^{\mathrm{SG}}(\boldsymbol{\mu}) \in W_N$ satisfies

$$\Theta(\boldsymbol{\mu}) a_L(u_N^{\mathrm{SG}}(\boldsymbol{\mu}), v) + \int_\Omega g(u_N^{\mathrm{SG}}(\boldsymbol{\mu}); \boldsymbol{\mu}) v \, \mathrm{d}\Omega = \ell(v) \quad \forall v \in W_N \tag{44}$$

DOI: 10.1002/nme

Unfortunately, the presence of *strong* non-linearity in g does not allow for the efficient *offline–online* procedure outlined in Section 3.2. As a result, although the dimension of system (44) is small, solving it is actually expensive: the evaluation of the integral $\int_\Omega g(u_N^{\mathrm{SG}}(\boldsymbol{\mu});\boldsymbol{\mu})v\,\mathrm{d}\Omega$ will scale as some power of $\mathcal{N}$. Therefore, it is somewhat disingenuous to interpret (44) as a reduced-order model.

We seek to develop a reduced-order approximation with an online evaluation cost *independent* of $\mathcal{N}$. Towards this goal, we compute a set of snapshots

$$\mathcal{S}_K^g \equiv \{\xi_k^g = g(u_h(\boldsymbol{\mu}_k^u);\boldsymbol{\mu}),\ 1\leqslant k\leqslant K\}$$

(Recall that the $u_h(\boldsymbol{\mu}_k)$ were already computed for all $\boldsymbol{\mu}_k^u \in S_K^u$.) We then construct $\{\mathbf{z}_m^g\}_{m=1}^M$ and $\{\psi_m^g\}_{m=1}^M$ by following the procedures described in Section 2. Then, for any given $w\in X_h$, we approximate $g(w;\boldsymbol{\mu})$ by $g_M^w = \sum_{m=1}^M g(w(\mathbf{z}_m^g);\boldsymbol{\mu})\psi_m^g$.

We may now replace $g(u_N^{\mathrm{SG}}(\boldsymbol{\mu});\boldsymbol{\mu})$—as required in our reduced-basis projection for $u_N^{\mathrm{SG}}(\boldsymbol{\mu})$—with $g_M^{u_{N,M}}(\mathbf{x};\boldsymbol{\mu})$. Our reduced-basis approximation is thus: given $\boldsymbol{\mu}\in\mathcal{D}$, we evaluate

$$s_{N,M}(\boldsymbol{\mu}) = \ell^O(u_{N,M}(\boldsymbol{\mu})) \tag{45}$$

where $u_{N,M}(\boldsymbol{\mu})\in W_N$ satisfies

$$\Theta(\boldsymbol{\mu})a_L(u_{N,M}(\boldsymbol{\mu}),v) + \int_\Omega g_M^{u_{N,M}}(\mathbf{x};\boldsymbol{\mu})v = \ell(v) \quad \forall v\in W_N \tag{46}$$

We expand our reduced-order approximation and coefficient-function approximation as

$$u_{N,M}(\boldsymbol{\mu}) = \sum_{j=1}^N u_{N,Mj}(\boldsymbol{\mu})\zeta_j$$

$$g_M^{u_{N,M}}(\mathbf{x};\boldsymbol{\mu}) = \sum_{m=1}^M g(u_{N,M}(\mathbf{z}_m^g);\boldsymbol{\mu})\psi_m^g = \sum_{m=1}^M g\left(\sum_{j=1}^N u_{N,Mj}(\boldsymbol{\mu})\zeta_j(\mathbf{z}_m^g);\boldsymbol{\mu}\right)\psi_m^g$$

Substituting these representations into (46) yields

$$\Theta(\boldsymbol{\mu})\mathbf{A}_N\mathbf{u}_{N,M}(\boldsymbol{\mu}) + \mathbf{C}_{N,M}\,g\big(\mathbf{D}_{M,N}\mathbf{u}_{N,M}(\boldsymbol{\mu});\boldsymbol{\mu}\big) = \mathbf{L}_N \tag{47}$$

where $\mathbf{A}_N\in\mathbb{R}^{N\times N}$, $\mathbf{C}_{N,M}\in\mathbb{R}^{N\times M}$, $\mathbf{D}_{M,N}\in\mathbb{R}^{M\times N}$, $\mathbf{L}_N\in\mathbb{R}^N$, and $\mathbf{L}_N^O\in\mathbb{R}^N$ are given by

$$\begin{aligned}
A_{Nij} &= a_L(\zeta_j,\zeta_i), \quad 1\leqslant i,\ \ j\leqslant N\\
C_{N,Mim} &= \int_\Omega \psi_m^g\zeta_i, \quad 1\leqslant i\leqslant N,\ \ 1\leqslant m\leqslant M\\
D_{M,Nmj} &= \zeta_j(\mathbf{z}_m^g), \quad 1\leqslant m\leqslant M,\ \ 1\leqslant j\leqslant N\\
L_{Ni} &= \ell(\zeta_i), \quad 1\leqslant i\leqslant N\\
L_{Ni}^O &= \ell^O(\zeta_i), \quad 1\leqslant i\leqslant N
\end{aligned} \tag{48}$$

These reduced-order matrices and vectors are parameter independent.

DOI: 10.1002/nme

To solve (47) for $\mathbf{u}_{N,M}(\boldsymbol{\mu})$, we may apply a Newton iterative scheme: given a current iterate $\bar{\mathbf{u}}_{N,M}(\boldsymbol{\mu})$ we must find an increment $\boldsymbol{\delta}\mathbf{u}_{N,M}(\boldsymbol{\mu})$ such that

$$(\Theta(\boldsymbol{\mu})\mathbf{A}_N + \bar{\mathbf{E}}_N(\boldsymbol{\mu}))\boldsymbol{\delta}\mathbf{u}_{N,M}(\boldsymbol{\mu}) = \mathbf{L}_N - \Theta(\boldsymbol{\mu})\mathbf{A}_N\bar{\mathbf{u}}_{N,M}(\boldsymbol{\mu}) - \mathbf{C}_{N,M}g(\mathbf{D}_{M,N}\bar{\mathbf{u}}_{N,M}(\boldsymbol{\mu}); \boldsymbol{\mu}) \tag{49}$$

Here, $\bar{\mathbf{E}}_N(\boldsymbol{\mu}) \in \mathbb{R}^{N\times N}$ must be calculated at every Newton iteration as

$$\bar{E}_{Nij}(\boldsymbol{\mu}) = \sum_{m=1}^{M} C_{N,Mim} g_1\left(\sum_{n=1}^{N} D_{M,Nmn}\bar{u}_{N,Mn}(\boldsymbol{\mu}); \boldsymbol{\mu}\right) D_{M,Nmj}, \quad 1 \leqslant i, \quad j \leqslant N \tag{50}$$

where $g_1(w; \boldsymbol{\mu})$ is the first derivative of g with respect to w. Note that calculating $\bar{\mathbf{E}}_N(\boldsymbol{\mu})$ has a cost of $O(MN^2)$.

Finally, we evaluate the reduced-order output as

$$s_{N,M}(\boldsymbol{\mu}) = (\mathbf{L}_N^O)^{\mathrm{T}}\mathbf{u}_{N,M}(\boldsymbol{\mu}) \tag{51}$$

Similar to that in Section 3.2, we can develop an efficient offline–online procedure for the rapid evaluation of $s_{N,M}(\boldsymbol{\mu})$ for each $\boldsymbol{\mu}$ in $\mathscr{D}$.

The operation count of the online stage is essentially the predominant Newton update component (49): at each Newton iteration, we first assemble the right-hand side and compute $\bar{\mathbf{E}}^N$ at cost $O(MN^2)$; we then form and invert the left-hand side (Jacobian) of (49) at cost $O(N^3)$. The online complexity depends only on N, M, and the number of Newton iterations; we thus recover $\mathscr{N}$ independence in the online stage.

4.3. Example 2: non-polynomial non-linearity

We consider a particular instantiation of our abstract statement in which

$$\Theta(\boldsymbol{\mu}) = \mu^1, \quad a_L(w, v) = \int_\Omega \nabla w \cdot \nabla v \, \mathrm{d}\Omega, \quad g(w; \boldsymbol{\mu}) = |w|^{\mu^2 - 1} w$$

and $\ell^O(v) = \ell(v) = \int_\Omega v \, \mathrm{d}\Omega$. Here, $\Omega =]0, 1[\times]0, 1[$, $\mathscr{D} = [0.01, 1] \times [1, 10] \in \mathbb{R}^2$, and $X = H_0^1(\Omega)$. Our model problem is well posed and becomes linear for $\mu^2 = 1$.

We present in Figure 8 two typical solutions obtained with a regular linear triangular FE approximation space X_h of dimension $\mathscr{N} = 10\,000$. We see that as μ^1 decreases and μ^2 increases, the solution develops a boundary layer. For the offline construction, we choose S_K^u as the nodes of a regular 12×12 grid over $\mathscr{D}$ and generate $\{\zeta_n\}_{n=1}^{N_{\max}}$, $\{\psi_m^g\}_{m=1}^{M_{\max}}$, and $\{\mathbf{z}_m^g\}_{m=1}^{M_{\max}}$ for $N_{\max} = 9$ and $M_{\max} = 10$. We plot the interpolation point set $\{\mathbf{z}_m^g\}_{m=1}^{M_{\max}}$ in Figure 9. We see that all the points lie in one-quarter of the domain due to the symmetry in the solution and that many points lie close to the boundary due to the presence of the boundary layer.

We now present numerical results for the reduced-order approximation. We show in Figure 10 the convergence of $\varepsilon_{\mathrm{ave,rel}}^u$ with respect to N and M. Here $\varepsilon_{\mathrm{ave,rel}}^u$ is the average relative error norm over the parameter sample $\Xi_{\mathrm{Test}} \subset \mathscr{D}$ of size 16×16. The reduced-order approximation converges very rapidly to the FE approximation. Furthermore, the error behavior is similar to that of the linear example of Section 3.3: the errors initially decrease, but then 'plateau' in N for a particular value of M; then increasing M reduces the errors further. This implies that for a given N, we can always choose M large enough so that the error induced by the coefficient-function approximation does

DOI: 10.1002/nme

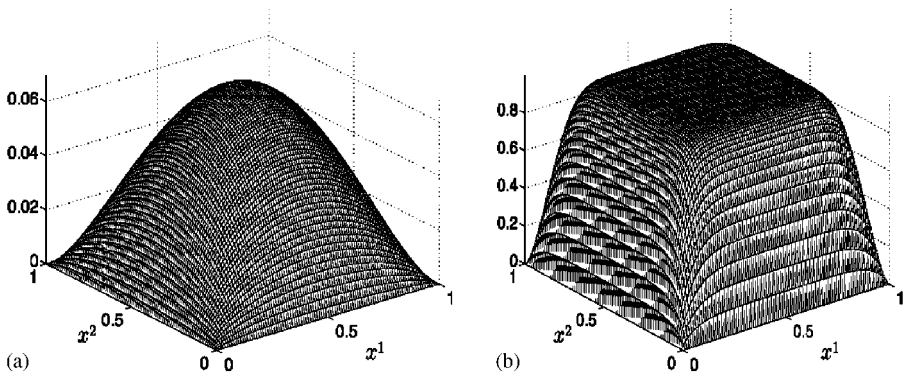

Figure 8. Numerical solutions at typical parameter points for the non-linear elliptic problem: (a) $\boldsymbol{\mu} = (1, 1)$ and (b) $\boldsymbol{\mu} = (0.01, 10)$.

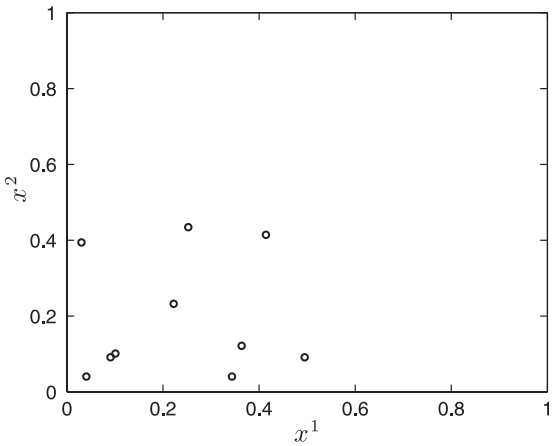

Figure 9. Distribution of the interpolation points $\{\mathbf{z}_m\}_{m=1}^{M_{\max}}$ over the physical domain Ω for $M_{\max} = 10$.

not affect the desired accuracy of our reduced-order approximation. Similar convergence results have also been observed for the RBA [33, 42].

We compare numerical results obtained with our ROA and with the standard Galerkin ROA. We tabulate in Table II the maximum relative error in the output over Ξ_{Test} and computational times as a function of N (for $M = 10$) for both the ROA and the SGA. Here, the computational times are normalized with respect to the time to compute $s_h(\boldsymbol{\mu})$ in the last column of Table II. We observe very high accuracy of the output approximation and significant computational savings: for a relative accuracy of less than 0.0002 ($N = 4$, $M = 10$), the online time to compute $s_{N,M}(\boldsymbol{\mu})$ is less than 1/30 000 the time to compute $s_h(\boldsymbol{\mu})$. In addition, thanks to fast convergence of the coefficient-function and reduced-order approximations and thanks to the $\mathcal{N}$*-independent* computational cost of the online stage, our reduced-order model is almost *four orders of magnitude* less expensive than the standard Galerkin reduced-order model, while yielding practically the same convergence and accuracy.

DOI: 10.1002/nme

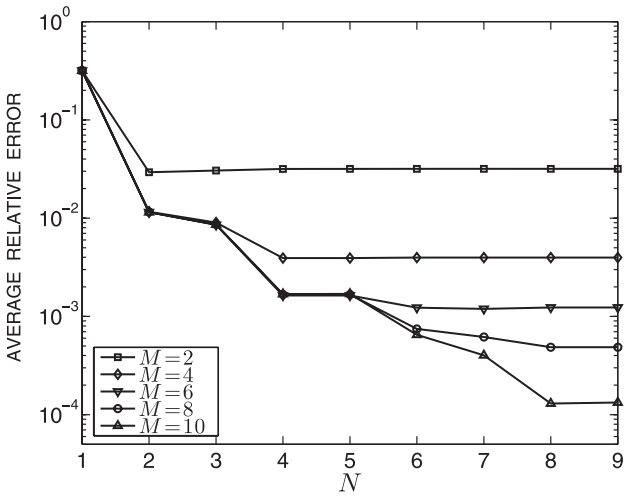

Figure 10. Average relative error norm $\varepsilon^{u}_{\text{ave,rel}}$ as a function of N and M for Example 2.

Table II. Numerical results for Example 2: maximum relative error in the output over Ξ_{Test} and normalized computational time as a function of N for the ROA (with $M = 10$) and the SGA.

		ROA		SGA		FEA
N	M	Maximum relative error	Online time	Maximum relative error	Online time	Computational time
1	10	2.02E−01	1.98E−05	2.02E−01	8.23E−01	1
2	10	2.33E−03	2.41E−05	2.33E−03	8.37E−01	1
3	10	2.25E−03	2.79E−05	2.25E−03	8.37E−01	1
4	10	1.90E−04	2.89E−05	1.90E−04	8.41E−01	1
5	10	1.81E−04	3.05E−05	1.81E−04	8.41E−01	1
6	10	4.14E−05	3.16E−05	4.14E−05	8.42E−01	1

Note: The computational times are normalized with respect to the computational time for $s_h(\boldsymbol{\mu})$.

5. NON-LINEAR CONVECTION–DIFFUSION EQUATIONS

In order to illustrate how our proposed approach can accommodate time-dependent problems, we consider a scalar non-linear convection–diffusion problem with only one parameter $\boldsymbol{\mu} \in \mathbb{R}$. We note that the approach can be easily extended to multi-parameter systems of PDEs. The essential new ingredient is the presence of time; we shall simply treat time as an additional, albeit special, parameter. We briefly describe the abstract formulation of non-linear convection–diffusion equations and then develop the associated reduced-order approximation. Finally, we discuss the results obtained using our approach and the RBA presented in [32, 33] for the two-dimensional Buckley–Leverett equation.

DOI: 10.1002/nme

5.1. Abstract problem formulation

We consider the following problem: for any given $\boldsymbol{\mu} \in \mathscr{D} \subset \mathbb{R}$, we find the output of interest as

$$s(\boldsymbol{\mu}, t) = \ell^O(u(\boldsymbol{\mu}, t)) \tag{52}$$

where $u(\boldsymbol{\mu}, t)$ is the solution to

$$m\left(\frac{\partial u(\boldsymbol{\mu}, t)}{\partial t}, v\right) - \int_\Omega \left(\frac{\partial f^1(u(\boldsymbol{\mu}, t))}{\partial x^1} + \frac{\partial f^2(u(\boldsymbol{\mu}, t))}{\partial x^2}\right) \mathrm{d}\Omega + \boldsymbol{\mu} a_L(u(\boldsymbol{\mu}, t), v) = 0 \quad \forall v \in X, \quad t \in (0, T] \tag{53}$$

with initial condition $u(\boldsymbol{\mu}, t) = u_0(\mathbf{x})$ and appropriate boundary condition. Here, $\boldsymbol{\mu}$ is a viscosity parameter varying in the parameter space $\mathscr{D} \in \mathbb{R}^1$; the fluxes, $(f^1(u(\boldsymbol{\mu}, t))$ and $f^2(u(\boldsymbol{\mu}, t)))$, are non-linear functions of the field variable $u(\boldsymbol{\mu}, t)$; a_L and m are parameter-independent bilinear forms; and ℓ^O is a linear functional. We note that the output and field variable are now functions of both the parameter $\boldsymbol{\mu}$ and time t.

We consider a discretization of (52)–(53) using the FE approximation and the second-order backward difference formula. We denote the number of time steps by J and let $\Delta t = T/J$. We further introduce a piecewise-linear FE approximation space X_h of *very* large dimension $\mathcal{N}$. Our FE discretization is thus: given any $\boldsymbol{\mu} \in \mathscr{D}$, for $j = 2, \ldots, J$, evaluate

$$s_h(\boldsymbol{\mu}, t^j) = \ell^O(u_h(\boldsymbol{\mu}, t^j)) \tag{54}$$

where $u_h(\boldsymbol{\mu}, t^j) \in X_h$ is the solution to

$$m(u_h(\boldsymbol{\mu}, t^j), v) - \frac{2}{3}\Delta t \int_\Omega \left(\frac{\partial f^1(u(\boldsymbol{\mu}, t))}{\partial x^1} + \frac{\partial f^2(u(\boldsymbol{\mu}, t))}{\partial x^2}\right) \mathrm{d}\Omega + \frac{2}{3}\Delta t \boldsymbol{\mu} a_L(u_h(\boldsymbol{\mu}, t^j), v) = \frac{4}{3} m(u_h(\boldsymbol{\mu}, t^{j-1}), v) - \frac{1}{3} m(u_h(\boldsymbol{\mu}, t^{j-2}), v) \quad \forall v \in X_h \tag{55}$$

subject to the initial condition $(u_h(\boldsymbol{\mu}, 0), v) = (u_0, v), \forall v \in X_h$. Note that marching (55) forward in time requires $u_h(\boldsymbol{\mu}, t^1)$ and that we use the Crank–Nicolson scheme to compute $u_h(\boldsymbol{\mu}, t^1)$ from $u_h(\boldsymbol{\mu}, 0)$. In essence, (55) yields a non-linear discrete system of equations which can be solved by Newton's method at each time step.

5.2. Reduced-order approximation

We introduce an approximation subspace $W_N = \mathrm{span}\{\zeta_1, \ldots, \zeta_N\}$, where the basis functions ζ_n are constructed using the POD procedure on a set of snapshots

$$\mathscr{S}_K^u \equiv \{\xi_k^u = u_h(\boldsymbol{\mu}_i^u, t^j), 1 \leqslant i \leqslant I, 1 \leqslant j \leqslant J, 1 \leqslant k \leqslant K = IJ\}$$

Here, $u_h(\boldsymbol{\mu}_i^u, t^j), 1 \leqslant j \leqslant J$, are the solutions to (55) at parameter points $\boldsymbol{\mu}_i, 1 \leqslant i \leqslant I$, belonging to the sample set $S_I^u = \{\boldsymbol{\mu}_1^u \in \mathscr{D}, \ldots, \boldsymbol{\mu}_I^u \in \mathscr{D}\}$.

DOI: 10.1002/nme

We next consider the standard Galerkin projection: for $\boldsymbol{\mu} \in \mathcal{D}$, we find

$$s_N^{\mathrm{SG}}(\boldsymbol{\mu}, t^j) = \ell^O(u_N^{\mathrm{SG}}(\boldsymbol{\mu}, t^j)), \quad 2 \leqslant j \leqslant J \tag{56}$$

where $u_N^{\mathrm{SG}}(\boldsymbol{\mu}, t^j) \in W_N$ satisfies

$$\begin{aligned} & m(u_N^{\mathrm{SG}}(\boldsymbol{\mu}, t^j), v) - \frac{2}{3}\Delta t \int_\Omega f^1(u_N^{\mathrm{SG}}(\boldsymbol{\mu}, t^j)) \frac{\partial v}{\partial x^1}\, \mathrm{d}\Omega \\ & \quad - \frac{2}{3}\Delta t \int_\Omega f^2(u_N^{\mathrm{SG}}(\boldsymbol{\mu}, t^j)) \frac{\partial v}{\partial x^2}\, \mathrm{d}\Omega + \frac{2}{3}\Delta t \boldsymbol{\mu} a_L(u_N^{\mathrm{SG}}(\boldsymbol{\mu}, t^j), v) \\ & = \frac{4}{3} m(u_N^{\mathrm{SG}}(\boldsymbol{\mu}, t^{j-1}), v) - \frac{1}{3} m(u_N^{\mathrm{SG}}(\boldsymbol{\mu}, t^{j-2}), v), \quad v \in W_N \end{aligned} \tag{57}$$

As pointed out earlier, this standard Galerkin reduced-order model is not efficient.

To recover efficiency, we develop the coefficient-function approximation for the non-linear terms in (57). Henceforth, we compute two sets of snapshots

$$\mathscr{F}_K^1 \equiv \{\xi_k^{f^1} = f^1(u_h(\boldsymbol{\mu}_i^u, t^j)), 1 \leqslant i \leqslant I, 1 \leqslant j \leqslant J, 1 \leqslant k \leqslant K = IJ\}$$

$$\mathscr{F}_K^2 \equiv \{\xi_k^{f^2} = f^2(u_h(\boldsymbol{\mu}_i^u, t^j)), 1 \leqslant i \leqslant I, 1 \leqslant j \leqslant J, 1 \leqslant k \leqslant K = IJ\}$$

from which we construct $\{\mathbf{z}_m^{f^1}\}_{m=1}^M$, $\{\psi_m^{f^1}\}_{m=1}^M$, and $\{\mathbf{z}_m^{f^2}\}_{m=1}^M$, $\{\psi_m^{f^2}\}_{m=1}^M$, respectively, following the procedures described in Section 2. (For notational simplification, we assume that the number of basis functions for both the coefficient-function expansions is the same.) Then for any given $w \in X_h$, we approximate $f^1(w)$ by $f_M^{1,w} = \sum_{m=1}^M f^1(w(\mathbf{z}_m^{f^1}))\psi_m^{f^1}$ and $f^2(w)$ by $f_M^{2,w} = \sum_{m=1}^M f^2(w(\mathbf{z}_m^{f^2}))\psi_m^{f^2}$.

Our reduced-order approximation is obtained by replacing $f^1(u_N^{\mathrm{SG}}(\boldsymbol{\mu}, t^j))$ and $f^2(u_N^{\mathrm{SG}}(\boldsymbol{\mu}, t^j))$ with $f_M^{1,u_{N,M}}(\boldsymbol{\mu}, t^j)$ and $f_M^{2,u_{N,M}}(\boldsymbol{\mu}, t^j)$: given any $\boldsymbol{\mu} \in \mathcal{D}$, for $j = 2, \ldots, J$, we evaluate

$$s_{N,M}(\boldsymbol{\mu}, t^j) = \ell^O(u_{N,M}(\boldsymbol{\mu}, t^j)) \tag{58}$$

where $u_{N,M}(\boldsymbol{\mu}, t^j) \in W_N$ satisfies

$$\begin{aligned} & m(u_{N,M}(\boldsymbol{\mu}, t^j), v) - \frac{2}{3}\Delta t \int_\Omega f_M^{1,u_{N,M}}(\boldsymbol{\mu}, t^j) \frac{\partial v}{\partial x^1}\, \mathrm{d}\Omega \\ & \quad - \frac{2}{3}\Delta t \int_\Omega f_M^{2,u_{N,M}}(\boldsymbol{\mu}, t^j) \frac{\partial v}{\partial x^2}\, \mathrm{d}\Omega + \frac{2}{3}\Delta t \boldsymbol{\mu} a_L(u_{N,M}(\boldsymbol{\mu}, t^j), v) \\ & = \frac{4}{3} m(u_{N,M}(\boldsymbol{\mu}, t^{j-1}), v) - \frac{1}{3} m(u_{N,M}(\boldsymbol{\mu}, t^{j-2}), v), \quad v \in W_N \end{aligned} \tag{59}$$

DOI: 10.1002/nme

Furthermore, we expand

$$u_{N,M}(\boldsymbol{\mu}, t^j) = \sum_{n=1}^{N} u_{N,Mn}(\boldsymbol{\mu}, t^j)\zeta_n$$

$$f_M^{1,u_{N,M}}(\boldsymbol{\mu}, t^j) = \sum_{m=1}^{M} f^1(u_{N,M}(\mathbf{z}_m^{f^1}; \boldsymbol{\mu}, t^j))\psi_m^{f^1} = \sum_{m=1}^{M} f^1\left(\sum_{n=1}^{N} u_{N,Mn}(\boldsymbol{\mu}, t^j)\zeta_n(\mathbf{z}_m^{f^1})\right)\psi_m^{f^1}$$

$$f_M^{2,u_{N,M}}(\boldsymbol{\mu}, t^j) = \sum_{m=1}^{M} f^2(u_{N,M}(\mathbf{z}_m^{f^2}; \boldsymbol{\mu}, t^j))\psi_m^{f^2} = \sum_{m=1}^{M} f^2\left(\sum_{n=1}^{N} u_{N,Mn}(\boldsymbol{\mu}, t^j)\zeta_n(\mathbf{z}_m^{f^2})\right)\psi_m^{f^2}$$

Substituting these representations into (59) and choosing $v = \zeta_{n'}, 1 \leqslant n' \leqslant N$, we immediately obtain the non-linear algebraic system: for $j = 2, \ldots, J$, $\mathbf{u}_{N,M}(\boldsymbol{\mu}, t^j) = [u_{N,M\ 1}(\boldsymbol{\mu}, t^j), \ldots, u_{N,MN}(\boldsymbol{\mu}, t^j)]^{\mathrm{T}} \in \mathbb{R}^N$ satisfies

$$\mathbf{M}_N \mathbf{u}_{N,M}(\boldsymbol{\mu}, t^j) - \tfrac{2}{3}\Delta t(\mathbf{C}_{N,M}^1 \mathbf{F}_M^1(\mathbf{u}_{N,M}(\boldsymbol{\mu}, t^j)) + \mathbf{C}_{N,M}^2 \mathbf{F}_M^2(\mathbf{u}_{N,M}(\boldsymbol{\mu}, t^j)))$$
$$+ \tfrac{2}{3}\Delta t \boldsymbol{\mu} \mathbf{A}_N \mathbf{u}_{N,M}(\boldsymbol{\mu}, t^j) = \tfrac{1}{3}\mathbf{M}_N(4\mathbf{u}_{N,M}(\boldsymbol{\mu}, t^{j-1}) - \mathbf{u}_{N,M}(\boldsymbol{\mu}, t^{j-2})) \tag{60}$$

with the initial vector $u_{N,Mn}(0) = (u_h(0), \zeta_n), 1 \leqslant n \leqslant N$, and with the coefficient vector $u_{N,M}(\boldsymbol{\mu}, t^1)$ being computed using the Crank–Nicolson scheme.

In the above, $\mathbf{M}_N \in \mathbb{R}^{N \times N}$ and $\mathbf{A}_N \in \mathbb{R}^{N \times N}$ are matrices with entries

$$M_{Nnn'} = m(\zeta_{n'}, \zeta_n), \quad A_{Nnn'} = a_L(\zeta_{n'}, \zeta_n) \tag{61}$$

for $1 \leqslant n, n' \leqslant N$; $\mathbf{C}_{N,M}^1 \in \mathbb{R}^{N \times M}$ and $\mathbf{C}_{N,M}^2 \in \mathbb{R}^{N \times M}$ are given by

$$C_{N,Mnm}^1 = \int_\Omega \psi_m^{f^1} \frac{\partial \zeta_n}{\partial x^1}, \quad C_{N,Mnm}^2 = \int_\Omega \psi_m^{f^2} \frac{\partial \zeta_n}{\partial x^2} \tag{62}$$

for $1 \leqslant n \leqslant N$, $1 \leqslant m \leqslant M$; and $\mathbf{F}_M^1(u_{N,M}(\boldsymbol{\mu}, t^j)) \in \mathbb{R}^M$ and $\mathbf{F}_M^2(u_{N,M}(\boldsymbol{\mu}, t^j)) \in \mathbb{R}^M$ are given by

$$\mathbf{F}_M^1(u_{N,M}(\boldsymbol{\mu}, t^j)) = f^1(\mathbf{D}_{M,N}^1 \mathbf{u}_{N,M}(\boldsymbol{\mu}, t^j)) \tag{63}$$

$$\mathbf{F}_M^2(u_{N,M}(\boldsymbol{\mu}, t^j)) = f^2(\mathbf{D}_{M,N}^2 \mathbf{u}_{N,M}(\boldsymbol{\mu}, t^j)) \tag{64}$$

where $\mathbf{D}_{M,N}^1 \in \mathbb{R}^{M \times N}$ and $\mathbf{D}_{M,N}^2 \in \mathbb{R}^{M \times N}$ are matrices with entries

$$D_{M,Nmn}^1 = \zeta_n(\mathbf{z}_m^{f^1}), \quad D_{M,Nmn}^2 = \zeta_n(\mathbf{z}_m^{f^2}) \tag{65}$$

for $1 \leqslant m \leqslant M, 1 \leqslant n \leqslant N$.

The non-linear algebraic system (60) can be readily solved using Newton's method at each time step for the coefficient vectors $\mathbf{u}_{N,M}(\boldsymbol{\mu}, t^j), 1 \leqslant j \leqslant J$. The reduced-order output is then calculated as

$$s_{N,M}(\boldsymbol{\mu}, t^j) = (\mathbf{L}_N^O)^{\mathrm{T}} \mathbf{u}_{N,M}(\boldsymbol{\mu}, t^j), \quad 1 \leqslant j \leqslant J \tag{66}$$

Here $\mathbf{L}_N^O \in \mathbb{R}^N$ is the output vector with entries $L_{Nn}^O = \ell^O(\zeta_n), 1 \leqslant n \leqslant N$. The offline-online procedure can be developed as follows.

DOI: 10.1002/nme

In the offline stage, we compute and store $\mathbf{u}_{N,M}(0)$, $\mathbf{A}_N$, $\mathbf{M}_N$, $\mathbf{C}^1_{N,M}$, $\mathbf{C}^2_{N,M}$, $\mathbf{D}^1_{M,N}$, $\mathbf{D}^2_{M,N}$, $\mathbf{L}^O_N$. In the online stage—for each new parameter value $\boldsymbol{\mu}$—we solve the non-linear system (60) at each time step for the coefficient vector $\mathbf{u}_{N,M}(\boldsymbol{\mu}, t^j)$ and evaluate the output $s_{N,M}(\boldsymbol{\mu}, t^j)$ with a computational cost (per Newton iteration per time step) of only $\mathcal{O}(MN^2+N^3)$.

5.3. *Example 3*: *Buckley–Leverett equation*

Our example is the two-dimensional Buckley–Leverett equation

$$\frac{\partial u}{\partial t}+\frac{\partial f^1(u)}{\partial x^1}+\frac{\partial f^2(u)}{\partial x^2}-\boldsymbol{\mu}\nabla^2 u=0 \quad \text{in } \Omega\times(0,T] \tag{67}$$

with initial condition $u_0(\mathbf{x})=\exp(-16((x^1)^2+(x^2)^2))$ and homogeneous boundary condition on $\partial\Omega$. Here $\Omega=]-1.5,1.5[^2$, $\boldsymbol{\mu}\in\mathscr{D}\equiv[0.05,0.1]$, $t\in(0,T]$ with $T=0.5$, and $f^1(u)$ and $f^2(u)$ are the fluxes that are non-linear functions of the field variable u,

$$f^1(u)=\frac{u^2}{u^2+(1-u)^2}, \quad f^2(u)=f_1(u)(1-5(1-u)^2)$$

The output of interest is the average of the field variable over the physical domain. The two-dimensional Buckley–Leverett equation is often used to describe two-phase flow in porous media with a gravitation pull in the x^1-direction.

The weak formulation is stated as: given $\boldsymbol{\mu}\in\mathscr{D}$, find $s(\boldsymbol{\mu},t)=\int_\Omega u(\boldsymbol{\mu},t)$, where $u(\boldsymbol{\mu},t)\in X=H^1_0(\Omega)\equiv\{v\in H^1(\Omega)|v|_{\partial\Omega}=0\}$ is the solution to

$$\int_\Omega \frac{\partial u}{\partial t}v\,\mathrm{d}\Omega-\int_\Omega f^1(u)\frac{\partial v}{\partial x^1}\,\mathrm{d}\Omega-\int_\Omega f^2(u)\frac{\partial v}{\partial x^2}\,\mathrm{d}\Omega+\boldsymbol{\mu}\int_\Omega \nabla u\cdot\nabla v=0 \quad \forall v\in X,\ \ t\in(0,T] \tag{68}$$

Our abstract statement (53)–(52) thus obtains for

$$m(w,v)=\int_\Omega wv\,\mathrm{d}\Omega, \quad a(w,v)=\int_\omega \nabla w\cdot\nabla v\,\mathrm{d}\Omega, \quad \ell^O(v)=\int_\Omega v\,\mathrm{d}\Omega$$

For the FE discretization, we use $\Delta t=0.02$ ($J=25$ time steps) and a uniform triangular piecewise-linear FE approximation space of dimension $\mathcal{N}=10000$. Figure 11 shows two typical solutions at time $t^j=10\Delta t$ for different values of $\boldsymbol{\mu}$. We see that when $\boldsymbol{\mu}=0.05$, the solution develops sharp gradient due to the strong effect of non-linearity. However, as $\boldsymbol{\mu}$ increases, the effect of non-linearity is dominated by the viscous effects, and the solution spreads out.

We now present the results obtained. For this purpose we define the average relative error in the solution as

$$\varepsilon^u_{\text{ave,rel}}=\text{mean}_{\boldsymbol{\mu}\in\Xi_{\text{Test}},1\leqslant j\leqslant J}\frac{\|u_h(\boldsymbol{\mu},t^j)-u_{N,M}(\boldsymbol{\mu},t^j)\|}{\|u_h(\boldsymbol{\mu},t^j)\|}$$

Here $\Xi_{\text{Test}}\subset\mathscr{D}$ is the parameter test sample of size 21. Figure 12 shows $\varepsilon^u_{\text{ave,rel}}$ as a function of N and M. We observe very rapid convergence of our reduced-order approximation and a similar convergence behavior as already seen in the previous examples. Note that our offline construction is based on $\mathscr{S}^u_K$, which consists of the initial solution $u_h(t^0)$ and 11×25 solutions corresponding to the parameter sample $S^u_I=\{\boldsymbol{\mu}^u_1,\ldots,\boldsymbol{\mu}^u_I\}$ of size $I=11$. We include $u_h(t^0)$ in $\mathscr{S}^u_K$ to render the error at time $t=0$ as small as possible.

DOI: 10.1002/nme

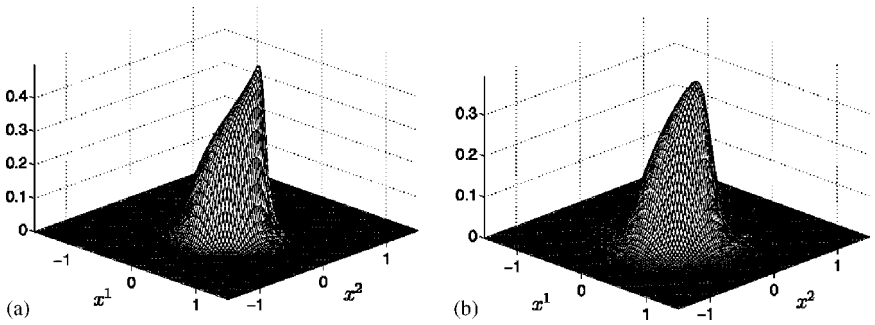

Figure 11. Numerical solutions for Example 3 at $t^j = 10\Delta t$: (a) $\boldsymbol{\mu} = 0.05$ and (b) $\boldsymbol{\mu} = 0.1$.

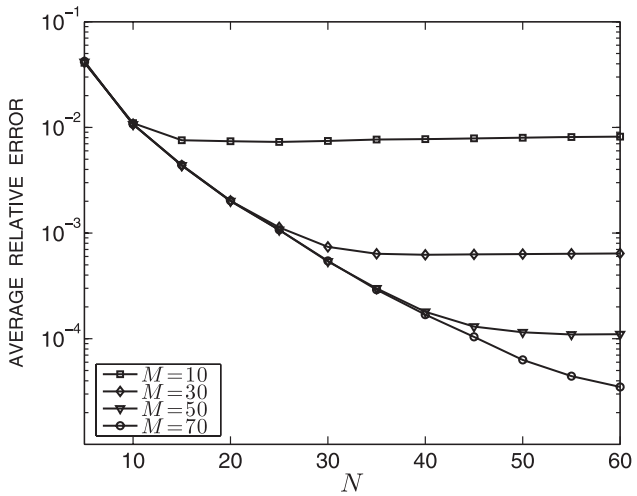

Figure 12. Average relative error $\varepsilon^{u}_{\text{ave,rel}}$ as a function of N and M for Example 3.

We next compare numerical results obtained with the proposed approach, the RBA of [32, 33], and the standard Galerkin ROA. In Table III, we present the maximum relative error in the output as a function of N and M for the ROA, RBA, and SGA. We first see that the three approaches yield uniform and rapid convergence of the reduced-order output approximation. However, the ROA results in smaller errors than the RBA for all values of N and M: for example, for the case of $(N, M) = (30, 40)$, the error is $9.51\text{E}-04$ for the ROA and is $3.10\text{E}-03$ for the RBA. This can be attributed to the choice of basis functions and interpolation points. First, for non-linear time-dependent problems, the interactions between the solution at different times are nicely captured by the global nature of the POD optimization; in this context, the greedy approach is not as successful as the POD approach. Second, the non-linearities and their interactions at different times are captured by the optimal selection of the BPIM, which provides a better accuracy for the coefficient-function approximation and the associated reduced-order model. In addition, we observe that the differences in the results obtained with the ROA and SGA are very small.

DOI: 10.1002/nme

Table III. Maximum relative error in the output for the reduced-order models generated by the proposed approach (ROA), the reduced-basis approach (RBA) of [32, 33], and the standard Galerkin projection approach (SGA) for different values of N and M for Example 3.

		ROA	RBA	SGA
N	M	Maximum relative error	Maximum relative error	Maximum relative error
10	20	2.08E−02	3.23E−02	2.08E−02
20	30	4.91E−03	8.31E−03	4.77E−03
30	40	9.51E−04	3.10E−03	9.42E−04
40	50	3.29E−04	5.26E−04	3.01E−04
50	60	9.91E−05	3.00E−04	9.62E−05

Table IV. Online computational times (normalized with respect to the computational time of the FE approximation) for different values of N and M for Example 3.

		ROA	RBA	SGA	FEA
N	M	Online time	Online time	Online time	Computational time
10	20	2.58E−05	2.58E−05	4.35E−01	1
20	30	3.16E−05	3.16E−05	4.37E−01	1
30	40	5.34E−05	5.34E−05	4.38E−01	1
40	50	8.50E−05	8.50E−05	4.40E−01	1
50	60	1.38E−04	1.38E−04	4.42E−01	1

Finally, in Table IV, we present the computational times to calculate the output as a function of N and M. Here the computational times are normalized with respect to the time to compute the FE approximation output shown in the last column of Table IV. For a relative accuracy of less than 0.1% (corresponding to $(N, M) = (30, 40)$ with the ROA and $(N, M) = (40, 50)$ with the RBA), the reduction in online response time for both the ROA and the RBA is more than four orders of magnitude compared with the SGA and the FEA. This is due to the dramatic dimension reduction provided by the Galerkin projection on the reduced-basis space and the coefficient-function approximation of the non-linear terms. We notice however that the offline computations are expensive since we must solve for the FE solutions over the parameter sample set. Hence, if a many-query context, or a clear demand for real-time response, can justify the offline cost, the proposed approach and the RBA presented in [32, 33] can be gainfully employed.

6. CONCLUSIONS

We have presented an efficient numerical approach for developing reduced-order models of nonaffine and non-linear parametrized PDEs. Although we discuss scalar problems and linear functionals, our approach can be easily extended to systems of equations and non-linear functionals. It is demonstrated through numerical examples that the approach provides computational savings of many orders of magnitude relative to both the FE approximation and standard Galerkin ROA. Compared with the RBA of [32, 33], the proposed method is found to produce better, more

DOI: 10.1002/nme

accurate, models at a slightly higher computational cost in the offline stage. Therefore, the proposed approach can be gainfully used to generate efficient low-order models of non-linear large-scale systems, which are of considerable interest in the parameter estimation, design, optimization, and control contexts.

In this paper, we have not considered the selection of snapshots and *a posteriori* error estimation. The accuracy, efficiency, and reliability of a reduced-order model depend crucially on the quality of the snapshot set for guaranteeing stable and rapid convergence, and *a posteriori* estimator for quantifying the error in the approximation process. While these important issues need to be addressed in the reduced-order modeling of parametrized PDEs, they remain very open and challenging especially for non-linear problems, the discussion of which is beyond the scope of this paper. Instead, we refer the reader to [41, 43] for a greedy algorithm for the judicious selection of snapshots and *a posteriori* error estimation procedures for linear and certain non-linear problems.

APPENDIX A: POD PROCEDURE

We describe the POD procedure to generate an orthonormal basis set $\{\varphi_n\}_{n=1}^N$ from *any* given set of linearly independent snapshots $\{\xi_k\}_{k=1}^K$. First, a two-point spatial correlation function is defined as

$$\mathscr{K}(\mathbf{x}, \mathbf{x}') = \frac{1}{K} \sum_{k=1}^{K} \xi_k(\mathbf{x}) \xi_k(\mathbf{x}') \tag{A1}$$

which accepts the following spectral decomposition:

$$\mathscr{K}(\mathbf{x}, \mathbf{x}') = \sum_{k=1}^{K} \lambda_k \varphi_k(\mathbf{x}) \varphi_k(\mathbf{x}') \tag{A2}$$

Here, the set of basis functions φ_k, $1 \leqslant k \leqslant K$, are ordered such that the associated eigenvalues

$$\lambda_k = \frac{1}{K} \sum_{l=1}^{K} (\varphi_k, \xi_l)_X^2 \tag{A3}$$

satisfy $\lambda_k \geqslant \lambda_{k+1}$.

Next, for a given $N < K$, the POD procedure consists in finding φ_n, $1 \leqslant n \leqslant N$, so as to maximize the captured energy

$$\max E_N = \sum_{n=1}^{N} \left(\frac{1}{K} \sum_{k=1}^{K} \left(\varphi_n, \xi_k\right)_X^2 \right) = \sum_{n=1}^{N} \lambda_n \tag{A4}$$

subject to the constraints $(\varphi_n, \varphi_{n'})_X = \delta_{nn'}$, $1 \leqslant n, n' \leqslant N$. The first few basis functions represent the main energy-containing structures in the snapshots, with their relative importance quantified by λ_n. It can be shown that problem (A4) amounts to solve the eigenfunction equation

$$(\mathscr{K}(\mathbf{x}, \mathbf{x}'), \varphi(\mathbf{x}'))_X = \lambda \varphi(\mathbf{x}) \tag{A5}$$

for the first N eigenfunctions.

DOI: 10.1002/nme

The method of snapshots [38] expresses a typical empirical eigenfunction $\varphi(\mathbf{x})$ as a linear combination of the ξ_k

$$\varphi(\mathbf{x}) = \sum_{k=1}^{K} a_k \xi_k(\mathbf{x}) \tag{A6}$$

Substituting this representation and (A1) into (A5), we immediately obtain

$$\mathbf{Ca} = \lambda \mathbf{a} \tag{A7}$$

where $\mathbf{C} \in \mathbb{R}^{K \times K}$ is given by $C_{ij} = (1/K)(\xi_i, \xi_j)_X$, $1 \leqslant i, j \leqslant K$. The eigenproblem (A7) can then be solved for the first N eigenvectors from which the POD basis functions φ_n, $1 \leqslant n \leqslant N$, are constructed using (A6).

The optimality of the POD basis can be shown by considering an arbitrary set of orthonormal basis functions, $\{v_n\}_{n=1}^N$, and demonstrating that the POD basis $\{\varphi_n\}_{n=1}^N$ minimizes

$$\min_{v_1,\ldots,v_N} \frac{1}{K} \left(\sum_{k=1}^{K} \inf_{\boldsymbol{\alpha}_N^k \in \mathbb{R}^N} \left\| \xi_k - \sum_{n=1}^{N} \alpha_{Nn}^k v_n \right\|_X^2 \right) \tag{A8}$$

Indeed, this minimization problem is equivalent to the maximization problem (A4), which in turn asserts the optimality of $\{\varphi_n\}_{n=1}^N$.

ACKNOWLEDGEMENTS

We would like to acknowledge Professor Anthony T. Patera of MIT and Professor Yvon Maday of the University Paris VI for their many invaluable contributions to this work. We would also like to thank Ms Hen Man of the Singapore-MIT Alliance for his helpful discussions. This work was supported by the Singapore-MIT Alliance.

REFERENCES

1. Almroth BO, Stern P, Brogan FA. Automatic choice of global shape functions in structural analysis. *AIAA Journal* 1978; **16**:525–528.
2. Noor AK, Peters JM. Reduced basis technique for nonlinear analysis of structures. *AIAA Journal* 1980; **18**(4): 455–462.
3. Fink JP, Rheinboldt WC. On the error behavior of the reduced basis technique for nonlinear finite element approximations. *Zeitschrift fuer Angewandte Mathematik and Mechanik* 1983; **63**:21–28.
4. Porsching TA. Estimation of the error in the reduced basis method solution of nonlinear equations. *Mathematics of Computation* 1985; **45**(172):487–496.
5. Gunzburger MD. *Finite Element Methods for Viscous Incompressible Flows*: *A Guide to Theory*, *Practice*, *and Algorithms*. Academic Press: Boston, MA, 1989.
6. Peterson JS. The reduced basis method for incompressible viscous flow calculations. *SIAM Journal on Scientific and Statistical Computing* 1989; **10**(4):777–786.
7. Prud'homme C, Rovas D, Veroy K, Maday Y, Patera AT, Turinici G. Reliable real-time solution of parametrized partial differential equations: reduced-basis output bound methods. *Journal of Fluids Engineering* 2002; **124**(1):70–80.
8. Deane A, Kevrekidis I, Karniadakis G, Orszag S. Low-dimensional models for complex geometry flows: application to grooved channels and circular cylinders. *Physics of Fluids A* 1991; **3**(10):2337–2354.
9. Berkooz G, Titi E. Galerkin projections and the proper orthogonal decomposition for equivariant equations. *Physics Letters A* 1993; **174**:94–102.

DOI: 10.1002/nme

10. Kunisch K, Volkwein S. Control of burgers' equation by a reduced order approach using proper orthogonal decomposition. *Journal on Optimization Theory and Applications* 1999; **102**:345–371.
11. Ravindran SS. A reduced-order approach for optimal control of fluids using proper orthogonal decomposition. *International Journal for Numerical Methods in Fluids* 2000; **34**:425–448.
12. Atwell JA, King BB. Proper orthogonal decomposition for reduced basis feedback controllers for parabolic equations. *Mathematical and Computer Modelling* 2001; **33**(1–3):1–19.
13. Willcox K, Peraire J. Balanced model reduction via the proper orthogonal decomposition. *AIAA Journal* 2002; **40**(11):2323.
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. Machiels L, Maday Y, Oliveira IB, Patera AT, Rovas D. Output bounds for reduced-basis approximations of symmetric positive definite eigenvalue problems. *Comptes Rendus de'l Academie des Sciences Paris*, *Séries I* 2000; **331**(2):153–158.
16. Maday Y, Patera AT, Turinici G. Global *a priori* convergence theory for reduced-basis approximation of single-parameter symmetric coercive elliptic partial differential equations. *Comptes Rendus de'l Academie des Sciences Paris*, *Séries I* 2002; **335**(3):289–294.
17. Veroy K, Rovas D, Patera AT. *A posteriori* error estimation for reduced-basis approximation of parametrized elliptic coercive partial differential equations: 'convex inverse' bound conditioners. *Control*, *Optimisation and Calculus of Variations* 2002; **8**:1007–1028. Special volume: a tribute to J.-L. Lions.
18. Grepl MA, Nguyen NC, Patera KAT, Liu GR. Certified rapid solution of parametrized partial differential equations for real-time applications. *Proceedings of the 2nd Sandia Workshop of PDE-Constrained Optimization*: *Towards Real-time and On-line PDE-constrained Optimization*, *SIAM Computational Science and Engineering Book Series*. SIAM: Philadelphia, PA, 2007; 197–212.
19. Grepl MA, Patera AT. Reduced-basis approximation for time-dependent parametrized partial differential equations. *Mathematical Modelling and Numerical Analysis* 2005; **39**:157–181.
20. Veroy K, Prud'homme C, Rovas DV, Patera AT. *A posteriori* error bounds for reduced-basis approximation of parametrized noncoercive and nonlinear elliptic partial differential equations (AIAA Paper 2003-3847). *Proceedings of the 16th AIAA Computational Fluid Dynamics Conference*, Orlando, Florida, 2003.
21. Nguyen NC, Veroy K, Patera AT. Certified real-time solution of parametrized partial differential equations. In *Handbook of Materials Modelling*, Yip S (ed.). Springer: Berlin, 2005; 1523–1559.
22. Veroy K, Patera AT. Certified real-time solution of the parametrized steady incompressible Navier–Stokes equations; rigorous reduced-basis *a posteriori* error bounds. *International Journal for Numerical Methods in Fluids* 2004; **47**:773–788.
23. Bui-Thanh T, Willcox K, Ghattas O, van Bloemen Waanders B. Goal-oriented, model-constrained optimization for reduction of large-scale systems. *Journal of Computational Physics* 2007; **224**:880–896.
24. Hinze M, Volkwein S. *Proper Orthogonal Decomposition Surrogate Models for Nonlinear Dynamical Systems*: *Error Estimates and Suboptimal Control*. Lecture Notes in Computational Science and Engineering, vol. 45. Springer: Berlin, 2005; **45**:261–306.
25. Graham WR, Peraire J, Tang KY. Optimal control of vortex shedding using low order models—Part I: Open-loop model development. *International Journal for Numerical Methods in Engineering* 1999; **44**:945–972.
26. Gunzburger M, Peterson J, Shadid J. Reduced-order modeling of time-dependent pdes with multiple parameters in the boundary data. *Computer Methods in Applied Mechanics and Engineering* 2007; **196**:1030–1047.
27. Bai ZJ. Krylov subspace techniques for reduced-order modeling of large-scale dynamical systems. *Applied Numerical Mathematics* 2002; **43**(1–2):9–44.
28. Rewienski M, White J. A trajectory piecewise-linear approach to model order reduction and fast simulation of nonlinear circuits and micromachined devices. *IEEE Transactions on Computer-Aided Design of Integrated Circuit and Systems* 2003; **22**:155–170.
29. Scherpen JMA. Balancing for nonlinear systems. *Systems and Control Letters* 1993; **21**:143–153.
30. Chen Y, White J. A quadratic method for nonlinear model order reduction. *Proceedings of the International Conference on Modelling and Simulation of Microsystems*, San Diego, CA, 2000; 477–480.
31. Phillips J. Projection-based approaches for model reduction of weakly nonlinear systems, time-varying systems. *IEEE Transactions on Computer-Aided Design of Integrated Circuit and Systems* 2003; **22**:171–187.
32. Barrault M, Maday Y, Nguyen NC, Patera AT. An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus de'l Academie des Sciences Paris*, *Séries I* 2004; **339**:667–672.

DOI: 10.1002/nme

33. Grepl MA, Maday Y, Nguyen NC, Patera AT. Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations. *Mathematical Modelling and Numerical Analysis* 2007; **41**:575–605.
34. Nguyen NC, Patera AT, Peraire J. A best points interpolation method for efficient approximation of parametrized functions. *International Journal of Numerical Methods in Engineering*, accepted.
35. Maday Y, Nguyen NC, Patera AT, Pau GSH. A general, multipurpose interpolation procedure: the magic points. *Constructive Approximation*, submitted.
36. Karhunen K. Zur spektraltheorie stochastischer prozesse. *Annales Academiae Scientiarum Fennicae* 1946; **37**:3–7.
37. Loeve MM. *Probability Theory*, Van Nostrand: Princeton, NJ, 1955.
38. Sirovich L. Turbulence and the dynamics of coherent structures. Part 1: Coherent structures. *Quarterly of Applied Mathematics* 1987; **45**(3):561–571.
39. Marquardt DW. An algorithm for least-squares estimation of nonlinear parameters. *SIAM Journal on Applied Mathematics* 1963; **11**(1):431–444.
40. Rozza G, Huynh D, Patera A. Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations: application to transport and continuum mechanics. *Archives of Computational Methods in Engineering*, submitted.
41. Nguyen NC. Reduced-basis approximations and a posteriori error bounds for nonaffine and nonlinear partial differential equations: application to inverse analysis. *Ph.D. Thesis*, Singapore-MIT Alliance, National University of Singapore, Singapore 2005.
42. Cancès E, LeBris C, Maday Y, Nguyen NC, Patera AT, Pau GSH. Feasibility and competitiveness of a reduced basis approach for rapid electronic structure calculations in quantum chemistry. *Proceedings of the Montreal Workshop for High-dimensional Partial Differential Equations in Science and Engineering*, vol. 41. CRM Proceedings Series. American Mathematical Society: Providence, RI, 2007; 15–47.
43. Grepl M. Reduced-basis approximations for time-dependent partial differential equations: application to optimal control. *Ph.D. Thesis*, Massachusetts Institute of Technology, Cambridge, MA, May 2005.

DOI: 10.1002/nme