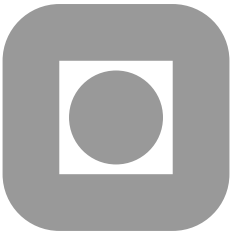
NORGES TEKNISK-NATURVITENSKAPELIGE
UNIVERSITET

An "hp" Certified Reduced Basis Method for Parametrized Parabolic Partial Differential Equations

by

Jens L. Eftang, Anthony T. Patera and Einar M. Rønquist

PREPRINT
NUMERICS NO. 9/2009



NORWEGIAN UNIVERSITY OF
SCIENCE AND TECHNOLOGY
TRONDHEIM, NORWAY

This report has URL
http://www.math.ntnu.no/preprint/numerics/2009/N9-2009.pdf
Address: Department of Mathematical Sciences, Norwegian University of Science and Technology, N-7491 Trondheim, Norway.

An "*hp*" Certified Reduced Basis Method for Parametrized Parabolic Partial Differential Equations

Jens L. Eftang, Anthony T. Patera, and Einar M. Rønquist

Abstract We extend previous work on a parameter multi-element "*hp*" certified reduced basis method for elliptic equations to the case of parabolic equations. A POD (in time) / Greedy (in parameter) sampling procedure is invoked both in the partitioning of the parameter domain ("*h*"-refinement) and in the construction of individual reduced basis approximation spaces for each parameter subdomain ("*p*"-refinement). The critical new issue is proper balance between additional POD modes and additional parameter values in the initial subdivision process. We present numerical results to compare the computational cost of the new approach to the standard ("*p*"-type) reduced basis method.

1 Introduction

The reduced basis (RB) method is a model-order reduction framework for rapid evaluation of functional outputs—such as surface temperatures or fluxes—for partial differential equations which depend on an input parameter vector—such as geometric factors or material properties. Given *any* parameter vector from a predefined parameter domain, the field variable is approximated as a Galerkin-optimal linear combination of accurately pre-computed "truth" finite element (FE) snapshots of the solution at judiciously selected parameter values [2, 6]; assuming that the field depends smoothly on the parameters, a RB approximation can be obtained with very few snapshots. Moreover, rigorous *a posteriori* upper bounds for the error in the RB approximation with respect to the truth discretization can be developed [4, 9].

Jens L. Eftang, Einar M. Rønquist
Department of Mathematical Sciences, Norwegian University of Science and Technology,
NO-7049 Trondheim, Norway. e-mail: eftang@math.ntnu.no, ronquist@math.ntnu.no

Anthony T. Patera
Department of Mechanical Engineering, Massachusetts Institute of Technology,
Cambridge, MA-02139, USA. e-mail: patera@mit.edu

The RB equation formation, solution, *and* error estimation can be made very efficient in the case of (perhaps approximate) "affine" parameter dependence through an offline-online procedure [8, 9]; the method is computationally attractive in two important engineering contexts—"real-time" and "many-query".

For many problems, the field variable may be quite different in different regions of the parameter domain, and hence a snapshot from one region may be of little value in approximating the solution in another region: the RB space is thus in some sense too large. In [3], an "*hp*" reduced basis method is introduced for linear elliptic equations: we adaptively subdivide the original parameter domain into smaller regions; we then build individual RB approximation spaces spanned by snapshots restricted to parameter vectors within each parameter subdomain. The RB approximation associated with any new parameter vector is then constructed as a linear (Galerkin) combination of snapshots from the parameter subdomain in which the new parameter vector resides. We thus expect that the dimension of the approximation space, and thus the online computational cost, to be very low: every basis function contributes significantly to the RB approximation. An alternative parameter-element reduced-order "interpolation" approach is introduced in [1].

In this paper, we extend the work in [3] to linear parabolic equations through a POD (in time) / Greedy (in parameter) sampling approach [5, 7]. This procedure determines the partition of the parameter domain *and* the construction of the individual RB approximation spaces for each subdomain. The elliptic machinery from [3] readily extends to the parabolic case since we only subdivide the parameter (and not the temporal) domain. The critical new issue is proper balance between additional POD modes and additional parameter values in the initial subdivision process.

Let $\Omega \subset \mathbf{R}^2$, define $L^2(\Omega) = \{v : \int_\Omega v^2 \, \mathrm{d}\Omega < \infty\}$, $H^1(\Omega) = \{v : \nabla v \in L^2(\Omega)\}$, $H^1_0(\Omega) = \{v \in H^1(\Omega) : v|_{\partial\Omega} = 0\}$, and introduce $X^e \equiv X^e(\Omega)$ such that $H^1_0(\Omega) \subseteq X^e(\Omega) \subset H^1(\Omega)$. The admissible parameter domain is $\mathscr{D} \subset \mathbf{R}^P$. For each $\mu \in \mathscr{D}$, $a(\cdot,\cdot;\mu)$ is a coercive and continuous bilinear form, $b(\cdot,\cdot;\mu)$ is an $L^2(\Omega)$ inner-product, and $f(\cdot;\mu)$ is a linear and bounded functional. We assume that a, b and f admit affine expansions in at most Q terms in the sense that (for example) $a(\cdot,\cdot;\mu) = \sum_{q=1}^{Q_a} \Theta^q(\mu) a^q(\cdot,\cdot)$, where the Θ^q are μ-dependent functions and the $a^q(\cdot,\cdot)$ are μ-independent bilinear forms, $1 \le q \le Q_a (\le Q)$; similar expansions in $Q_b \le Q$ and $Q_f \le Q$ terms apply for b and f, respectively. Let $\bar{\mu} \in \mathscr{D}$ be a fixed "reference parameter", and denote the symmetric part of a by a_s; we then define the X-inner-product and X-norm by $a_s(\cdot,\cdot;\bar{\mu})$ and $\|\cdot\|_X = (a_s(\cdot,\cdot;\bar{\mu}))^{1/2}$, respectively.

We shall consider problems already discretized in time with the Euler Backward (EB) method. Let $[0,T]$ be the time interval and introduce $K+1$ discrete time-values $t^k = \Delta t k, 0 \le k \le K$, where $\Delta t = T/K$ is the step-size. Our "exact" (hence ^{e}) problem then reads: Given any $\mu \in \mathscr{D}$, find $u^e(t^k,\mu) \in X^e$, $1 \le k \le K$, such that

$$\frac{1}{\Delta t} b\big(u^e(t^k;\mu) - u^e(t^{k-1};\mu), v; \mu\big) + a\big(u^e(t^k;\mu), v; \mu\big) = f(v;\mu), \quad \forall v \in X^e; \quad (1)$$

we assume zero initial conditions, $u^e(t^0;\mu) = 0$. The output of interest can now be evaluated as a functional of the field variable; in this paper however, for simplicity

of exposition, we consider no particular outputs of interest. Note that since our problem is linear time-invariant (LTI), we may also readily treat time-dependent (offline unknown) control functions through an impulse approach [4].

The RB approximation will be built upon truth FE approximations to the "exact" solution; let $X \equiv X^{\mathcal{N}}(\Omega) \subset X^e(\Omega)$ denote a FE space of dimension $\mathcal{N}$. We shall assume that X is rich enough that the error between the truth and "exact" solutions is in practice negligible. The truth discretization then reads: Given any $\mu \in \mathcal{D}$, find $u^k(\mu) \equiv u(t^k;\mu) \in X$, $1 \le k \le K$, such that

$$\frac{1}{\Delta t} b\big(u^k(\mu) - u^{k-1}(\mu), v; \mu\big) + a\big(u^k(\mu), v; \mu\big) = f(v;\mu), \quad \forall v \in X; \tag{2}$$

for initial condition $u^0(\mu) = 0$.

In Section 2, we formulate the "*hp*" RB method for parabolic problems, review the POD/greedy sampling procedure from [5], and discuss the new parameter domain partitioning approach. In Section 3, we present numerical results and discuss the computational cost of the new approach relative to the standard method.

2 The "*hp*" Reduced Basis Method

Reduced Basis Approximation Assume that $\mathcal{D}$ is divided into M parameter subdomains $\mathcal{V}_m \subset \mathcal{D}$, $1 \le m \le M$. The partitioning procedure is briefly reviewed below; see [3] for further details. Each subdomain has an associated set of nested RB approximation spaces $X_{N,m} \subset X$, $1 \le N \le N_{\max,m}$, (where $\dim(X_{N,m^*}) = N$) constructed by the POD/Greedy sampling procedure. The parameter domain partitioning, the POD/Greedy sampling, and the computation of the truth snapshots are all effected in an offline computational stage; this stage may be rather expensive in terms of computational cost, but is carried out only once as a pre-processing step.

Given any new $\mu \in \mathcal{D}$ in the online stage, the algorithm first determines which subdomain $\mathcal{V}_{m^*} \subset \mathcal{D}$ contains μ, and then selects the associated approximation space X_{N,m^*} from a database of offline-constructed spaces. Once m^* ($1 \le m^* \le M$) is determined, the RB approximation reads: Given any N and any $\mu \in \mathcal{D}$, find $u_N^k(\mu) \equiv u_{\hat{N},m^*}^k(\mu) \in X_{\hat{N},m^*}$, $1 \le k \le K$, such that

$$\frac{1}{\Delta t} b(u_N^k(\mu) - u_N^{k-1}(\mu), v; \mu) + a(u_N^k(\mu), v; \mu) = f(v;\mu), \quad \forall v \in X_{\hat{N},m^*}, \tag{3}$$

subject to $u_N^0(\mu) = 0$; here $\hat{N} = \min\{N, N_{\max,m^*}\}$. The offline-online decoupling and associated computational procedures are explained in detail in [8, 9]. In particular, the online computational cost and storage requirements are independent of $\mathcal{N}$—the dimension of the truth FE space—thanks to our "affine" assumption on the parameter dependence.

A posteriori **Error Estimation** For each $\mu \in \mathscr{D}$, denote by $\alpha_{\rm LB}(\mu) < \alpha(\mu) = \inf_{v\in X} a(v,v;\mu)/\|v\|_X$ a lower bound for the coercivity constant of $a(\cdot,\cdot;\mu)$. We then define the "energy norm" for $w^k \in X$,

$$|||w^k||| = \left(b(w^k,w^k;\mu) + \Delta t \sum_{k'=1}^{k} a_s(w^k,w^k;\mu) \right)^{1/2}, \quad 1 \le k \le K. \tag{4}$$

Given an RB approximation for $\mu \in \mathscr{V}_m \subset \mathscr{D}$, $u_N^k(\mu)$, $1 \le k \le K$, we write the residual as $r_N^k(v;\mu) = f(v;\mu) - b(u_N^k(\mu) - u_N^{k-1}(\mu),v;\mu)/\Delta t - a(u_N^k(\mu),v;\mu)$ and denote by $\varepsilon_N^k(\mu) = \sup_{v\in X} r_N^k(v;\mu)/\|v\|_X$ its dual X-norm. The energy norm of the RB error $e_N^k(\mu) = u^k(\mu) - u_N^k(\mu)$, $1 \le k \le K$, is bounded by

$$\Delta_N^k(\mu) \equiv \left(\Delta t \sum_{k'=1}^{k} (\varepsilon_N^k(\mu))^2 \Big/ \alpha_{\rm LB}(\mu) \right)^{1/2} \ge |||e_N^k(\mu)|||. \tag{5}$$

For a proof of (5) and the associated (offine-online) computational procedures for the dual norm of the residuals and the coercivity lower bound, see [4, 8, 9].

POD/Greedy Sampling In order to determine the parameter domain partitioning ("h"-refinement) and, associated with each subdomain, individual RB approximation spaces ("p"-refinement), we invoke the POD/Greedy sampling procedure introduced in [5] (see also [7]). We first describe in this section the standard "p"-type POD/Greedy procedure applied to the entire parameter domain $\mathscr{D}$. We then consider in the next section the application of the POD/Greedy procedure in the "hp" context.

Let the function POD($\{w^1 \in X, 1 \le k \le K\}, R$) return $R (\le K)$ X-orthonormal functions $\{\chi^i \in X, 1 \le i \le R\}$ such that $\mathscr{P}_R = \text{span}\{\chi^i, 1 \le i \le R\}$ satisfies the optimality property

$$\mathscr{P}_R = \arg \inf_{Y \subset \text{span}\{w^k, 1\le k\le K\}} \left(\frac{1}{K} \sum_{k=1}^{K} \inf_{w\in Y} \|w^k - w\|_X \right)^{1/2}. \tag{6}$$

To obtain the set $\{\chi^i, 1 \le i \le R\}$—the first R *POD modes* of span$\{w^1,\ldots,w^K\}$—we first solve the eigenvalue problem $C\psi^i = \lambda^i \psi^i$ for $(\psi^i \in \mathbf{R}^K, \lambda^i \in \mathbf{R})$ associated with the R largest eigenvalues of C, where $C_{ij} = (w_i, w_j)_X$, $1 \le i,j \le K$; we then compute $\chi^i = \sum_{k=1}^K \psi_k^i w^k$ for $1 \le i \le R$.

Let $\Xi \subset \mathscr{D}$ be a (typically very rich) finite training sample over $\mathscr{D}$. We initialize the POD/Greedy(R,L) algorithm by choosing (randomly, say) $\mu^* \in \mathscr{D}$ and setting $X_0 = \{0\}$, $N = 1$. Then, while $N \le L$, we first compute the projection error $e_{N,\text{proj}}^k(\mu^*) = u^k(\mu^*) - \text{proj}_{X_{N-1}}(u^k(\mu^*))$, $1 \le k \le K$, where $\text{proj}_{X_N} w$ denotes the X-orthogonal projection of $w \in X$ onto X_N. Next, we expand the RB space with the R first POD-modes, $X_N \leftarrow X_N \oplus \text{span}\{\text{POD}(\{e_{N,\text{proj}}^k(\mu^*), 1 \le k \le K\}, R)\}$, and set $N \leftarrow N + R$. Finally, the next parameter vector is chosen greedily over Ξ based on the *a posteriori* error estimator at the final time; $\mu^* \leftarrow \arg\max_{\mu\in\Xi} \Delta_N^K(\mu)$.

Parameter Domain Partitioning Since we subdivide only the parameter (and not the temporal) domain, the "hp" reduced basis framework described in detail for

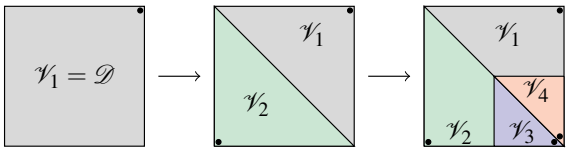


Fig. 1 Hierarchical partitioning of the parameter domain based on proximity to greedily chosen parameter anchor points.

elliptic problems in [3] also applies to the parabolic context of this paper. The "parabolic" algorithm developed here differs from the "elliptic" algorithm of [3] in the definition of the error bound and in particular in the choice of the parameter sampling procedure: care must be taken to properly balance additional POD modes and additional parameter values in partitioning the parameter domain.

The parameter domain partition is determined in the offline stage. We start from the original domain $\mathscr{D}$, choose $\mu^* = \mu_0^* \in \mathscr{D}$, and perform the POD/Greedy(R,L) algorithm over $\mathscr{D}$ with $R = R_1 \geq 1$ and $L = R_1$ (such that we perform only a single POD). We denote the resulting (nested) approximation spaces as $X_{N,1}$, $1 \leq N \leq R_1$, and the next parameter vector as μ_1^*. Based on proximity (e.g. Euclidian distance) to the two parameter *anchor points* μ_0^* and μ_1^*, we can now divide $\mathscr{D}$ into two new subdomains $\mathscr{V}_0 \subset \mathscr{D}$, $\mathscr{V}_1 \subset \mathscr{D}$, respectively. We now repeat the procedure within each subdomain for $\mu^* = \mu_0^*$ and $\mu^* = \mu_1^*$ as the initial parameter vectors, respectively; note that one of the two "child" subdomains inherits the parameter anchor point, and thus the associated approximation space, from its "parent." In Fig. 1, we illustrate the partitioning algorithm with two levels of refinement; we proceed recursively until the error bound at the final time is less than $\varepsilon_{\text{tol}}^1$ (over train samples) over each subdomain.

We must comment on the tuning parameter R_1, which is crucial to the convergence of the "*h*"-refinement stage of the algorithm. In particular, R_1 must be chosen large enough that the RB error bound associated with the (R_1-dimensional) RB approximation at the final time is less than $\varepsilon_{\text{tol}}^1$ in a neighborhood of μ^*. Otherwise, the procedure would not converge since the tolerance would not be reached. Note it is not sufficient that the tolerance is satisfied only at μ^*, since then the tolerance might not be satisfied at any point arbitrarily close to μ^*, and the partitioning algorithm might yield arbitrarily small subdomains.

In particular, we shall require that the error bound associated with the RB approximation of $u^K(\mu^*)$ based on R_1 POD modes is less than $\varepsilon_{\text{tol}}^1/\rho_1$: $\rho_1 > 1$ ensures that the RB error bound is smaller than $\varepsilon_{\text{tol}}^1$ in a neighborhood of μ^*; the refinement algorithm will then converge since eventually the subdomain containing μ^* will be included in this neighborhood. Note that choosing $\rho_1 > 1$ too small would lead to a large number of subdomains, while large ρ_1 will require more POD modes to be included in the RB space; in the limit $\rho_1 \to \infty$, we would need to include all K POD modes in the RB space in order to achieve a zero RB error (bound) at μ^* at the final time—as in the elliptic case, there would thus perforce be a neighborhood around μ^* where the RB error bound would be very small and in particular less than $\varepsilon_{\text{tol}}^1$.

It remains to determine R_1 automatically. Towards that end, we note that the "POD energy norm" in (6) is very similar to the energy norm defined in (4): since

$e^k_{1,\mathrm{proj}}(\mu^*) = u^k(\mu^*)$, the *POD error*—the square root of the sum of the eigenvalues λ^i, $i = R_1+1,\ldots,K$—is closely related to the RB error bound; in fact, the POD error multiplied by the coercivity lower bound $\alpha_{\mathrm{LB}}(\mu^*)$ is a lower bound for the RB error bound at μ^*. As an initial guess, we thus choose R_1 such that the associated POD error at μ^* is less than $\varepsilon^1_{\mathrm{tol}}\alpha_{\mathrm{LB}}(\mu^*)/\rho_1\rho_2$, where we choose $\rho_2 \geq 1$ to compensate for the fact that the POD error is smaller than the RB error bound. Next, we compute the RB error bound associated with the RB approximation of $u^k(\mu^*)$, $1 \leq k \leq K$, based on R_1 POD modes: if the error bound is smaller than $\varepsilon^1_{\mathrm{tol}}/\rho_1$, we conclude that R_1 is sufficiently large; if not, we successively set $R_1 \leftarrow R_1+1$, increase the number of POD modes, and compute a new RB error bound—until the tolerance is satisfied.

This "h"-refinement results in a total of M subdomains $\mathcal{V}_m \subset \mathcal{D}$, $1 \leq m \leq M$. The next step is "p"-refinement: we expand the approximation spaces associated with each subdomain $\mathcal{V}_m$, $m = 1,\ldots,M$, by application of the POD/Greedy(R,L) sampling procedure for $R = R_2$ and $L \geq R_1$ "specified"; in actual practice, we terminate the POD/Greedy in subdomain m for $L \equiv N_{\max,m}(\varepsilon^2_{\mathrm{tol}})$ such that the error bound is less than a second tolerance $\varepsilon^2_{\mathrm{tol}} < \varepsilon^1_{\mathrm{tol}}$ (over the training sample) over the subdomain—the final approximation spaces $X_{N,m}$, $1 \leq N \leq N_{\max,m}$, $1 \leq m \leq M$, will thus in general have different dimensions. We typically choose $R_2 = 1$; note that $R_2 > 1$ will lead to improved offline performance but worse online performance.

We now turn to the online stage, in which for every new $\mu \in \mathcal{D}$, the algorithm first determines which approximation space to invoke, and then computes the RB approximation and associated RB error bound. Note that since the subdomains are constructed hierarchically based on proximity to the parameter anchor points associated with each subdomain, we can determine the subdomain containing μ in an efficient (typically negligible) $\mathcal{O}(\log_2 M)$-operations binary search. In particular, once $\mathcal{V}_{m^*} \subset \mathcal{D}$ containing μ is found, we solve (3) for the RB space X_{N,m^*}, and compute the error bound (5); the total cost is $\mathcal{O}(N^3 + Q^2N^2)$, as described in more detail shortly.

3 A Convection-Diffusion Model Problem

We now apply the "hp" RB method to a convection-diffusion model problem parametrized by the angle and magnitude of the specified velocity field: let $\mu = (\mu_1,\mu_2)$ (hence $P = 2$ parameters) and define $\mathbf{V}(\mu) = [\mu_2\cos\mu_1, \mu_2\sin\mu_1]$; we shall consider $\mu \in \mathcal{D} = [0,\pi]\times[1,10]$. The physical domain is $\Omega = \{(x,y) : x^2+y^2 < 2\}$; the final time is $T = 1$ and the timestep is $\Delta t = 0.05$ such that $K = 20$. The "exact" field $\tilde{u}^e(t,\mu)$ satisfies $(\tilde{u}^e(t^k;\mu) - \tilde{u}^e(t^{k-1};\mu))/\Delta t - \nabla^2\tilde{u}^e(t^k;\mu) + \mathbf{V}(\mu)\cdot\nabla\tilde{u}^e(t^k;\mu) = 10$, $1 \leq k \leq K$; we apply homogeneous Dirichlet boundary conditions; we consider an *inhomogeneous* initial condition (hence the tilde) $\tilde{u}^e(t^0) = g$, where g satisfies $-\nabla^2 g = 10$ in Ω.

We now reduce our equation to the desired form (1). We first write $\tilde{u}^e = u^e + g$ such that u^e now satisfies homogeneous initial conditions. We then define $b(w,v;\mu) = \int_\Omega wv\,\mathrm{d}\Omega$, $a(w,v;\mu) = \int_\Omega \nabla w\cdot\nabla v\,\mathrm{d}\Omega + \int_\Omega \big(\mathbf{V}(\mu)\cdot\nabla w\big)v\,\mathrm{d}\Omega$, and

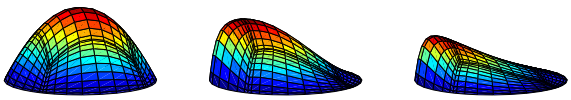


Fig. 2 Example solutions for the convection-diffusion problem at $t = 0, 0.1, 0.25$ for the parameter value $\mu = (\pi, 10)$.

$f(v;\mu) = 10\int_\Omega v\,\mathrm{d}\Omega - a(g,v;\mu)$; note that a_s is μ-independent and in fact we may choose $\alpha_{\mathrm{LB}}(\mu) = 1$. Thus u^e satisfies (1) (and homogeneous initial conditions) with $Q_a = 3$, $Q_b = 1$, and $Q_f = 4$. We next introduce a truth space $X \equiv X^{\mathcal{N}}(\Omega)$: five spectral elements each of polynomial order 10. Figure 2 depicts the truth solution at $t = 0, 0.1, 0.25$ for the parameter value $\mu = (\pi, 10)$. As the parameters vary, the solution changes dramatically—a good candidate for "*hp*" treatment.

We now apply the POD/Greedy procedure to partition $\mathcal{D}$ into M parameter subdomains; the resulting "*hp*" RB approximation can then be written in the form (3). In Fig. 3, we show the partition of the parameter domain for $M = 97$ and $M = 2258$ subdomains corresponding to $\varepsilon_{\mathrm{tol}}^1 = 5$ and $\varepsilon_{\mathrm{tol}}^1 = 1$, respectively; we choose $\rho_1 = 2$ and $\rho_2 = 1$; we also report, for each of the two partitions shown, the maximum of the error bound over the training samples over all subdomains as a function of N; we include the standard "*p*"-type RB approximation ($M = 1$) as well. Clearly, with smaller subdomains we need fewer basis functions for each approximation space.

We summarize in Table 1 for different error tolerances $\varepsilon_{\mathrm{tol}}^2$ the offline and online performance of the "*hp*" approach relative to that of the standard RB method. We report the number of truth solves (effectively parameters visited in the POD/Greedy); the number of operations for online evaluation of $u_N^k(\mu)$, $1 \le k \le K$, and $\Delta_N^k(\mu)$; and the online storage. The values in the table are based on the *theoretical* operation count and storage requirement. For N basis functions the online operation count (for our LTI system) is roughly $2N^3/3 + 2KN^2$ operations for the RB solution (and,

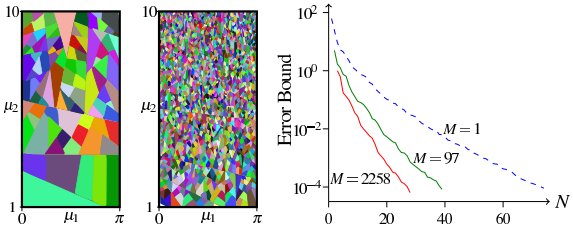


Fig. 3 Partition of $\mathcal{D}$ into $M = 97$ and $M = 2258$ subdomains ($\varepsilon_{\mathrm{tol}} = 5$ and $\varepsilon_{\mathrm{tol}}^1 = 1$, respectively); maximum error bound as a function of the RB approximation space dimension.

Table 1 Offline and online effort relative to standard ("*p*"-type) RB method for the two partitions $M = 97$ subdomains *(left)* and $M = 2258$ subdomains *(right)* for different tolerances $\varepsilon_{\mathrm{tol}}^2$.

Tolerance, $\varepsilon_{\mathrm{tol}}^2$	10^{-2}	10^{-3}	10^{-4}
Truth solves	39.2	40.7	40.6
Online $u_N^k(\mu)$	0.20	0.22	0.21
Online $\Delta_N^k(\mu)$	0.25	0.28	0.28
Online storage	16.7	17.7	17.5

Tolerance, $\varepsilon_{\mathrm{tol}}^2$	10^{-2}	10^{-3}	10^{-4}
Truth solves	597	660	659
Online $u_N^k(\mu)$	0.08	0.09	0.09
Online $\Delta_N^k(\mu)$	0.11	0.13	0.14
Online storage	166	200	197

in pratice, output), and $\mathcal{O}(Q^2N^2 + KN^2)$ operations for the RB error bound (see [4, 8] for details); we neglect the $\mathcal{O}(QN^2)$ cost of forming the RB system and the $\mathcal{O}(\log_2 M)$ cost of finding the correct subdomain. For each space (subdomain) the storage requirement is $\mathcal{O}(Q^2N^2)$.

The new method is admittedly more expensive in terms of the *offline* cost—the number of truth solves. However, significant computational savings are achieved in the *online* computation of the RB solution and RB error bound; note that for modest Q the costs of the RB solution and RB error bound are comparable. For "real-time" or "many-query" applications the online cost is often our main concern and the "hp" approach is thus very attractive. Note however, that "p"-type refinement plays a crucial role in controlling the offline cost, in particular in higher parameter dimensions.

Future work on the "hp" approach will focus on quadratically nonlinear problems: in these cases the online operation count is $\mathcal{O}(N^4)$ and thus computational performance can greatly benefit from the (further) dimension reduction afforded by the "hp" approach.

Acknowledgements We acknowledge helpful discussions with D. Knezevic, N. C. Nguyen, and S. Boyaval. The work has been supported by NTNU and AFOSR Grant FA 9550-07-1-0425.

References

1. D. Amsallem, J. Cortial, and C. Farhat. On-demand CFD-based aeroelastic predictions using a database of reduced-order bases and models. In: 47th AIAA Aerospace Sciences Meeting (2009).
2. B. O. Almroth, P. Stern, and F. A. Brogan. Automatic choice of global shape functions in structural analysis. *AIAA J.*, **16**, 525-528, 1978.
3. J. L. Eftang, A. T. Patera, and E. M. Rønquist. An *hp* Reduced Basis Method for Parametrized Elliptic Partial Differential Equations. In preparation, 2009.
4. M. A. Grepl and A. T. Patera. *A posteriori* error bounds for reduced-basis approximations of parametrized parabolic partial differential equations. *M2AN*, **39**, 157–181, 2005.
5. B. Haasdonk and M. Ohlberger. Reduced basis method for finite volume approximations of parametrized linear evolution equations. *M2AN Math. Model. Numer. Anal.*, **42**, 277–302, 2008.
6. A. K. Noor and J. M. Peters. Reduced basis technique for nonlinear analysis of structures. *AIAA J.*, **18**, 455-462, 1980.
7. D. J. Knezevic and A. T. Patera. A Certified Reduced Basis Method for the Fokker-Planck Equation of Dilute Polymeric Fluids: FENE Dumbbells in Extensional Flow. *SIAM J. Sci. Comput.*, (submitted 2009).
8. N. C. Nguyen, G. Rozza, D. B. P. Huynh, and A. T. Patera. Reduced Basis Approximation and *A Posteriori* Error Estimation for Paramtrized Parabolic PDEs; Application to Real-Time Bayesian Parameter Estimation. In: Biegler, et al. Computational Methods for Large Scale Inverse Problems and Uncertainty Quantification. John Wiley and Sons (2009).
9. G. Rozza, D. B. P. Huynh, and A. T. Patera. Reduced Basis Approximation and *A Posteriori* Error Estimation for Affinely Parametrized Elliptic Coercive Partial Differential Equations. *Arch. Comput. Methods Eng.*, **15**, 229–275, 2008.