REDUCED BASIS METHODS BASED UPON ADAPTIVE SNAPSHOT COMPUTATIONS

KRISTINA STEIH AND KARSTEN URBAN

ABSTRACT. We use asymptotically optimal *adaptive* numerical methods (here specifically a wavelet scheme) within the offline phase of the Reduced Basis Method (RBM). The resulting parameter-dependent discretizations do not permit the standard RB "truth space", but allow for error estimation of the RB approximation with respect to the exact solution of the considered parameterized partial differential equation. The evaluation of the estimators is also performed adaptively. This RBM with adaptive offline computations is analyzed. We show that multiple selection of snapshots may occur and devise strategies to avoid this. Numerical experiments for stationary and instationary problems show potential and challenges of this approach.

1. INTRODUCTION

Reduced Basis Methods (RBMs) have nowadays become a widely accepted and used tool for realtime and/or multi-query simulations of parameterized partial differential equations (PPDEs). By using an offline-online decomposition, the main idea is to use a high fidelity, detailed, but costly numerical solver offline to compute approximations to the PPDEs for certain parameter values. The selection of these parameters is done by an error estimator which is efficiently computable and thus allows one to determine the 'worst' parameters out of a possible rich so-called training set. For those 'bad' parameters, the high fidelity model is used in order to determine approximations, so-called snapshots. These few snapshots form the reduced basis which is then capable to produce approximations for any new parameter value extremely rapidly (online). The error estimator can also be used online in order to certify this RB approximation. Both the variety of applications and the amount of recent results in RBMs go well beyond the scope of such an introduction.

The success of the RBM relies on the assumption that the high fidelity model in the offline phase is sufficiently accurate *for all* parameters. The same discretization is used for all snapshots and all error estimators. This may have some possible drawbacks: (1) If this high fidelity model is not accurate enough, also the RBapproximation cannot be good. (2) The other extreme is that a sufficiently accurate approximation for all possible parameters may require a high fidelity model whose dimension is too large even for an offline phase. (3) The error estimate usually

Date: May 5, 2014.

²⁰¹⁰ Mathematics Subject Classification. 35B10, 41A30, 41A63, 65N30, 65Y20.

Key words and phrases. Reduced Basis Method, adaptivity, wavelets.

This work has partly been supported by the Deutsche Forschungsgemeinschaft within the Research Training Group (Graduiertenkolleg) GrK1100 Modellierung, Analyse und Simulation in der Wirtschaftsmathematik at Ulm University.

controls the difference to the high fidelity solution, not w.r.t. the exact solution of the PPDE (with one recent exception in [25] to be discussed below).

On the other hand, there are adaptive numerical methods available that guarantee an approximation of the exact solution of a PDE within a preselected tolerance. Such methods can be based upon finite element or wavelet discretizations, [6, 7, 18, 23]. We use such an adaptive method (we choose wavelets) for computing both snapshots and error estimators in the offline phase. This offers some features that we think are of interest, namely: (a) We use different discretizations for each parameter allowing for a minimal amount of work for any chosen parameter. (b) We can bound the RB error w.r.t. the exact solution of the PPDE.

Using adaptivity (or different discretizations) in the offline phase implies some additional sophistication of the method, at least from the conceptual point of view. The question arises under which circumstances such adaptivity might pay off. It is known e.g. from [6] that adaptive methods show faster convergence rates if the Besov regularity of the solution in a certain scale exceeds the Sobolev regularity. For the offline RB setting this means that the regularity of the solution with respect to the parameter is of crucial importance. If one single discretization is sufficient for approximating the solution $u(\mu)$ well enough for all possible parameters μ , then adaptivity seems not to make sense. On the other hand, if $u(\mu)$ significantly differs w.r.t. μ , a joint discretization may be too fine. This is e.g. the case if $u(\mu)$ has parameter-dependent local effects. Our numerical examples are guided by these considerations.

The remainder of this paper is organized as follows. In Section 2, we review the main facts of the 'classical' Reduced Basis Method. We set the framework for PPDEs and collect those facts that are needed here. Section 3 is devoted to the use of adaptive methods for the generation of the reduced basis in the offline phase. At this point, we only require the availability of a certain adaptive solver **SOLVE** and do not specify which specific method is used. We have used an Adaptive Wavelet Galerkin Method (AWGM). Since it is not absolutely necessary to describe the precise adaptive method within the RB framework in Section 3, we have collected a brief description of the AWGM in Appendix A. Finally, in Section 4 we describe numerical experiments for two different examples, namely heat conduction in a thermal block with several local heat sources and time-dependent convection-diffusion-reaction using a space-time variational formulation.

2. Reduced Basis Methods (RBMs)

In order to highlight differences and challenges of using adaptively computed basis functions within the Reduced Basis Method (RBM), it makes sense to briefly review 'standard' RBMs.

2.1. Parameterized Partial Differential Equations (PPDEs). Let $\Omega \subset \mathbb{R}^n$ be a bounded domain on which we consider function spaces $\mathcal{X} = \mathcal{X}(\Omega)$, $\mathcal{Y} = \mathcal{Y}(\Omega)$ arising from a variational formulation of a partial differential equation. Denoting by $\mathcal{D} \subset \mathbb{R}^P$ the set of parameters, this means that we consider a differential operator $\mathcal{B} : \mathcal{X} \times \mathcal{D} \to \mathcal{Y}'$ resp. a bounded bilinear form $b : \mathcal{X} \times \mathcal{Y} \times \mathcal{D} \to \mathbb{R}$, where $b(w, v; \mu) := \langle \mathcal{B}[w; \mu], v \rangle_{\mathcal{Y}' \times \mathcal{Y}}$ for $w \in \mathcal{X}, v \in \mathcal{Y}$ and $\mu \in \mathcal{D}$. In particular, we assume the existence of constants $\gamma(\mu) \leq \gamma^{\text{UB}} < \infty$ such that

(2.1)
$$b(w,v;\mu) \le \gamma(\mu) \|w\|_{\mathcal{X}} \|v\|_{\mathcal{Y}}, \qquad w \in \mathcal{X}, v \in \mathcal{Y}.$$

For a given $g(\mu) \in \mathcal{Y}'$, the problem is then to find a $u = u(\mu) \in \mathcal{X}$ such that $\mathcal{B}[u(\mu);\mu] = g(\mu)$ in \mathcal{Y}' , or, in variational form

(2.2)
$$b(u(\mu), v; \mu) = f(v; \mu) \quad \forall v \in \mathcal{Y},$$

where $f(v; \mu) := \langle g(\mu), v \rangle_{\mathcal{Y}' \times \mathcal{Y}}$. It is required that a numerical solver for (2.2) is available, e.g. finite volume, finite element or wavelet methods.

We assume that (2.2) is well-posed for all $\mu \in \mathcal{D}$, which is equivalent to the socalled *Nečas condition* on $b(\cdot, \cdot; \mu)$, [17, 18], i.e. there exist *inf-sup constants* $\beta(\mu)$ and a lower bound β_{LB} such that

(2.3)
$$\beta(\mu) = \inf_{w \in \mathcal{X}} \sup_{v \in \mathcal{Y}} \frac{b(w, v; \mu)}{\|w\|_{\mathcal{X}} \|v\|_{\mathcal{Y}}} = \inf_{w \in \mathcal{Y}} \sup_{u \in \mathcal{X}} \frac{b(w, v; \mu)}{\|w\|_{\mathcal{X}} \|v\|_{\mathcal{Y}}} \ge \beta_{\mathrm{LB}} > 0$$

for all $\mu \in \mathcal{D}$.

Remark 2.1. (a) It is worth mentioning that (2.2) includes elliptic problems, where e.g. $\mathcal{X} = \mathcal{Y} = H_0^1(\Omega)$ (or other boundary conditions), $b(\cdot, \cdot; \mu)$ being coercive with constant $\alpha(\mu) > 0$, as well as parabolic initial value problems in space-time formulation, i.e. with the Bochner spaces $\mathcal{X} = W_0(0,T;V) := \{u \in L_2(0,T;V) :$ $u_t \in L_2(0,T;V'), u(0) = 0 \in H\}, \mathcal{Y} = L_2(0,T;V)$, so that $\mathcal{X} \subsetneq \mathcal{Y}$, and also timeperiodic problems, see also Section 4 below.

(b) Instead of a space-time formulation for a parabolic initial value problem, one could also use a standard time-stepping scheme. There are corresponding RBMs available for such problems [11, 12]. In principle, our subsequent findings can be extended also to those settings, but in order to keep notations simple, we restrict ourselves to (2.2).

2.2. Some Basics on 'Classical' RBMs. Any numerical scheme for the solution of (2.2) involves a discretization of \mathcal{X} , \mathcal{Y} . In a standard RB setting these finitedimensional discrete spaces, the so-called *truth spaces*, are denoted by $\mathcal{X}^{\mathcal{N}} \subset \mathcal{X}$, $\mathcal{Y}^{\mathcal{N}} \subset \mathcal{Y}$.^a Then, the following Galerkin projection is considered:

(2.4) Find
$$u^{\mathcal{N}}(\mu) \in \mathcal{X}^{\mathcal{N}}$$
: $b(u^{\mathcal{N}}(\mu), v; \mu) = f(v; \mu) \quad \forall v \in \mathcal{Y}^{\mathcal{N}}.$

Often, $\mathcal{X}^{\mathcal{N}}$, $\mathcal{Y}^{\mathcal{N}}$ are spanned by *local basis functions* such as finite elements or wavelets and their dimension $\mathcal{N} = \dim(\mathcal{X}^{\mathcal{N}}) = \dim(\mathcal{Y}^{\mathcal{N}})^{\mathrm{b}}$ is usually large, so that solving (2.4) repeatedly for many different parameters would be too costly or realtime computations would be impossible.

Remark 2.2 (Fixed discretization). We stress that in the standard RB setting, the spaces $\mathcal{X}^{\mathcal{N}}$, $\mathcal{Y}^{\mathcal{N}}$ are a-priorily *fixed* and are the *same* for all parameters $\mu \in \mathcal{D}$. Moreover, it is assumed that the discretization error $\|u(\mu) - u^{\mathcal{N}}(\mu)\|_{\mathcal{X}}$ is negligibly small for all $\mu \in \mathcal{D}$. Thus, typical RBMs view $u^{\mathcal{N}}(\mu)$ as 'truth', which means e.g. that all error estimates are typically w.r.t. $u^{\mathcal{N}}(\mu)$ and do *not* take $u(\mu)$ into account. Just recently a first paper appeared introducing error bounds w.r.t. $u(\mu)$ in a specific case, using techniques, however, that do not seem to be applicable in a general framework, [25].

^aWe always use calligraphic symbols for high-(even ∞)-dimensional spaces.

^bFor simplicity, we assume that trial and test spaces are of the same dimension. Otherwise, one would need to use a least squares approach.

The idea behind (standard) RBMs is the construction of low-dimensional spaces $X_N^{\mathcal{N}} \subset \mathcal{X}^{\mathcal{N}}, Y_N^{\mathcal{N}} \subset \mathcal{Y}^{\mathcal{N}}, N \ll \mathcal{N},^{c}$ from so-called *snapshots*, i.e. solutions of (2.4) for selected parameters, i.e.

(2.5)
$$X_N^{\mathcal{N}} := \operatorname{span}\{u^{\mathcal{N}}(\mu_i), i = 1, \dots, N\} =: \operatorname{span}\{\zeta_i^{\mathcal{N}}, i = 1, \dots, N\},\$$

and $Y_N^{\mathcal{N}} := \operatorname{span}\{\xi_i^{\mathcal{N}}, i = 1, \dots, N\}$ is such that the N-dimensional reduced problem

Find
$$u_N^{\mathcal{N}}(\mu) \in X_N^{\mathcal{N}}$$
: $b(u_N^{\mathcal{N}}(\mu), v; \mu) = f(v; \mu) \quad \forall v \in Y_N^{\mathcal{N}}$

is stable. Stability in the discrete setting is ensured by the fulfillment of a *discrete inf-sup* condition [1], i.e.,

(2.6)
$$\inf_{w_N \in X_N^{\mathcal{N}}} \sup_{v_N \in Y_N^{\mathcal{N}}} \frac{b(w_N, v_N; \mu)}{\|w_N\|_{\mathcal{X}} \|v_N\|_{\mathcal{Y}}} \ge \bar{\beta} > 0,$$

with $\bar{\beta}$ independent of N as $N \to \infty$. We abbreviate

$$(2.7) S_N := \{\mu_1, \dots, \mu_N\}$$

as the set of parameter values corresponding to the snapshots. The system $\{\zeta_i^N, i = 1, \ldots, N\}$ may arise by orthonormalization of the snapshots.

The inf-sup condition (2.3) gives rise to rigorous a posteriori *error bounds*, i.e. \mathcal{N} -independent quantities $\Delta_N^{\mathcal{N}}(\mu)$ with

(2.8)
$$\|e_N^{\mathcal{N}}(\mu)\|_{\mathcal{X}} := \|u^{\mathcal{N}}(\mu) - u_N^{\mathcal{N}}(\mu)\|_{\mathcal{X}} \le \Delta_N^{\mathcal{N}}(\mu) = \frac{\|r_N^{\mathcal{N}}(\cdot;\mu)\|_{\mathcal{Y}'}}{\beta(\mu)},$$

where $r_N^{\mathcal{N}}(\cdot;\mu): \mathcal{Y}^{\mathcal{N}} \to \mathbb{R}$ is the residual with respect to the reduced solution, i.e.

$$r_N^{\mathcal{N}}(v;\mu) := f(v;\mu) - b(u_N(\mu),v;\mu), \qquad \forall v \in \mathcal{Y}^{\mathcal{N}}.$$

The involved dual norms $\|r_N^{\mathcal{N}}(\cdot;\mu)\|_{\mathcal{Y}'}$ are computed with the help of the *Riesz* representations $\hat{r}_N^{\mathcal{N}}(\mu) \in \mathcal{Y}^{\mathcal{N}}$ solving

(2.9)
$$\left(\hat{r}_N^{\mathcal{N}}(\mu), v\right)_{\mathcal{Y}} = r_N^{\mathcal{N}}(v; \mu) \quad \forall v \in \mathcal{Y}^{\mathcal{N}}.$$

The efficient \mathcal{N} -independent computation of $\|r_N^{\mathcal{N}}(\cdot; \mu)\|_{\mathcal{Y}'} = \|\widehat{r}_N^{\mathcal{N}}(\mu)\|_{\mathcal{Y}}$ relies on an offline-online decomposition, see §2.4 below. It is easily seen that this error bound is also effective. In fact, we have for any $v \in \mathcal{Y}^{\mathcal{N}}$ that

$$\begin{aligned} (\hat{r}_{N}^{\mathcal{N}}(\mu), v)_{\mathcal{Y}} &= r_{N}^{\mathcal{N}}(v; \mu) = f(v; \mu) - b(u_{N}^{\mathcal{N}}(\mu), v; \mu) = b(u^{\mathcal{N}}(\mu) - u_{N}^{\mathcal{N}}(\mu), v; \mu) \\ &= b(e_{N}^{\mathcal{N}}(\mu), v; \mu). \end{aligned}$$

Inserting $v = \hat{r}_N^{\mathcal{N}}(\mu)$ and using the boundedness (2.1) yields

$$\|\widehat{r}_N^{\mathcal{N}}(\mu)\|_{\mathcal{Y}}^2 = b(e_N^{\mathcal{N}}(\mu), \widehat{r}_N^{\mathcal{N}}(\mu); \mu) \le \gamma(\mu) \|e_N^{\mathcal{N}}(\mu)\|_{\mathcal{X}} \|\widehat{r}_N^{\mathcal{N}}(\mu)\|_{\mathcal{Y}},$$

hence $\|\widehat{r}_{N}^{\mathcal{N}}(\mu)\|_{\mathcal{Y}} \leq \gamma(\mu) \|e_{N}^{\mathcal{N}}(\mu)\|_{\mathcal{X}}$, so that the error estimator $\Delta_{N}^{\mathcal{N}}(\mu) = \frac{\|\widehat{r}_{N}^{\mathcal{N}}(\mu)\|_{\mathcal{Y}}}{\beta(\mu)}$ and the error $\|e_{N}^{\mathcal{N}}(\mu)\|_{\mathcal{X}}$ are in fact equivalent:

(2.10)
$$\|e_N^{\mathcal{N}}(\mu)\|_{\mathcal{X}} \le \Delta_N^{\mathcal{N}}(\mu) \le \frac{\gamma(\mu)}{\beta(\mu)} \|e_N^{\mathcal{N}}(\mu)\|_{\mathcal{X}}$$

^cLow-dimensional spaces are denoted by usual (non calligraphic) symbols.

Remark 2.3. We point out (for later reference in §3.4 below) that $u_N^{\mathcal{N}}(\mu) = u^{\mathcal{N}}(\mu)$ for all $\mu \in S_N$, i.e., snapshots are reproduced by the standard RBM. In fact, we have Petrov-Galerkin orthogonality, i.e., $b(u^{\mathcal{N}}(\mu) - u_N^{\mathcal{N}}(\mu), v_N; \mu) = 0$ for all $v_N \in Y_N^{\mathcal{N}}$. Since $u^{\mathcal{N}}(\mu) \in X_N^{\mathcal{N}}$ for $\mu \in S_N$, we have that $e_N^{\mathcal{N}}(\mu) = u^{\mathcal{N}}(\mu) - u_N^{\mathcal{N}}(\mu) \in X_N^{\mathcal{N}}$ and then (2.6) yields

$$\bar{\beta} \| e_N^{\mathcal{N}}(\mu) \|_{\mathcal{X}} \le \sup_{v_N \in Y_N^{\mathcal{N}}} \frac{b(e_N^{\mathcal{N}}(\mu), v_N; \mu)}{\|v_N\|_{\mathcal{Y}}} = 0,$$

i.e., $u^{\mathcal{N}}(\mu) = u_N^{\mathcal{N}}(\mu)$ for all $\mu \in S_N$.

2.3. Basis Construction via the Greedy Algorithm. The choice of the RB basis functions $\zeta_i^{\mathcal{N}}$, i = 1, ..., N, i.e. the selection of the corresponding parameter values μ_1, \ldots, μ_N , is often done using a Greedy algorithm: Given μ_1, \ldots, μ_n , n < N, the next parameter value μ_{n+1} is chosen as

$$\mu_{n+1} = \arg \max_{\mu \in \Xi_{\text{train}}} \Delta_N^{\mathcal{N}}(\mu),$$

where $\Xi_{\text{train}} \subset \mathcal{D}$ is a finite *training set*. The key point for the efficiency of this approach is the fact that the greedy selection is done w.r.t. the \mathcal{N} -independent error estimator. Only for the chosen parameter values μ_1, \ldots, μ_N the (expensive) truth has to be computed. The corresponding algorithm is displayed in Algorithm 1. Note that this procedure is also called *weak Greedy training*, in contrast to an (inefficient) *strong* Greedy where the true error $\|e_N^{\mathcal{N}}(\mu)\|_{\mathcal{X}}$ is used in line 2. More precisely, if $\|e_N^{\mathcal{N}}(\mu)\|_{\mathcal{X}} \leq \gamma \Delta_N^{\mathcal{N}}$ for some $\gamma > 0$, then Algorithm 1 is called *weak Greedy with parameter* γ .

Algorithm 1 $[X_N^N] =$ Greedy $[tol, N_{max}, \Xi_{train}]$

1: for $N = 0, 1, ..., N_{\max}$ do 2: Choose $\mu^* := \operatorname{argmax}_{\mu \in \Xi_{\operatorname{train}}} \Delta_N^{\mathcal{N}}(\mu)$. 3: if $\Delta_N^{\mathcal{N}}(\mu^*) < \operatorname{tol}$ then return 4: $N \leftarrow N + 1$. 5: Compute 'truth' snapshot $u^{\mathcal{N}}(\mu^*)$, update basis: $X_N^{\mathcal{N}} = X_{N-1}^{\mathcal{N}} \cup \{u^{\mathcal{N}}(\mu^*)\}$. 6: end for

Remark 2.4. As an alternative to the Greedy algorithm one could determine μ^* by nonlinear optimization, [4, 24]. If feasible, this approach avoids a training set.

2.4. Offline-online Decomposition. A crucial assumption for the efficiency of the RBM (in particular the efficient computation of $u_N^{\mathcal{N}}(\mu)$ and of $\Delta_N^{\mathcal{N}}(\mu)$) is that the bilinear form and the right-hand side are affine in the parameter, i.e.

(2.11)
$$b(u,v;\mu) = \sum_{q=1}^{Q_b} \theta_b^{(q)}(\mu) \, b^{(q)}(u,v), \qquad f(v;\mu) = \sum_{q=1}^{Q_f} \theta_f^{(q)}(\mu) \, f^{(q)}(v).$$

Techniques like the Empirical Interpolation Method (EIM) [2] can construct an approximation of such an affine decomposition if assumption (2.11) is not met. Affine forms as in (2.11) enable an efficient offline-online decomposition of the calculations in the following sense: The parameter-independent components of the linear system, namely $\mathbf{B}_{N}^{(q)} := \left[b^{(q)}(\zeta_{i}^{\mathcal{N}},\xi_{j}^{\mathcal{N}})\right]_{i,j=1,\ldots,N}, q = 1,\ldots,Q_{b}$, and $\mathbf{F}_{N}^{(q)} :=$

 $\left[f^{(q)}(\xi_j^{\mathcal{N}})\right]_{j=1,\ldots,N}, q=1,\ldots,Q_f$, can be precomputed (offline) so that the assembly and solution of the reduced system $\mathbf{B}_N(\mu)\mathbf{u}_N(\mu) = \mathbf{F}_N(\mu)$ with

(2.12)
$$\mathbf{B}_{N}(\mu) := \sum_{q=1}^{Q_{b}} \theta_{b}^{(q)}(\mu) \, \mathbf{B}^{(q)}, \ \mathbf{F}_{N}(\mu) := \sum_{q=1}^{Q_{f}} \theta_{f}^{(q)}(\mu) \, \mathbf{F}^{(q)},$$

for a new parameter $\mu \notin S_N$ then only involves N-dimensional matrix-vector products and can be done online (with complexity independent of \mathcal{N}). Since $\mathbf{B}_N(\mu) \in \mathbb{R}^{N \times N}$ is usually densely populated, the linear system (2.12) for

$$u_N^{\mathcal{N}}(\mu) = \sum_{i=1}^N u_N^{(i)}(\mu) \,\zeta_i^{\mathcal{N}}, \qquad \mathbf{u}_N(\mu) = [u_N^{(i)}(\mu)]_{i=1,\dots,N},$$

can be solved with $\mathcal{O}(N^3)$ operations – independent of $\mathcal{N} \gg N$.

.

Also the error estimate can be computed online-efficient (independent of \mathcal{N}). This can again be achieved by using (2.11) as follows: The problems

(2.13a) Find
$$\hat{b}_{\mathcal{N}}^{q,i} \in \mathcal{Y}^{\mathcal{N}}$$
: $(\hat{b}_{\mathcal{N}}^{q,i}, v)_{\mathcal{Y}} = -b^{(q)}(\zeta_i^{\mathcal{N}}, v) \quad \forall v \in \mathcal{Y}^{\mathcal{N}}$

(2.13b) Find
$$\hat{f}_{\mathcal{N}}^{q'} \in \mathcal{Y}^{\mathcal{N}}$$
: $(\hat{f}_{\mathcal{N}}^{q'}, v)_{\mathcal{Y}} = f^{(q')}(v) \qquad \forall v \in \mathcal{Y}^{\mathcal{N}},$

are solved offline (with complexity $\mathcal{O}(\mathcal{N})$) for all $q = 1, \ldots, Q_b, i = 1, \ldots, N$, $q' = 1, \ldots, Q_f$, and the inner products

$$\left[\left(\hat{f}_{\mathcal{N}}^{q}, \hat{f}_{\mathcal{N}}^{q'} \right)_{\mathcal{Y}} \right]_{q,q'=1}^{Q_{f}}, \left[\left(\hat{f}_{\mathcal{N}}^{q}, \hat{b}_{\mathcal{N}}^{q',i} \right)_{\mathcal{Y}} \right]_{q,q'=1}^{Q_{f},Q_{b}}, \left[\left(\hat{b}_{\mathcal{N}}^{q,i}, \hat{b}_{\mathcal{N}}^{q',i'} \right)_{\mathcal{Y}} \right]_{q,q'=1}^{Q_{b}}, \quad i,i'=1,\ldots,N,$$

are computed and stored. During the online phase, the parameter-dependent norm $||r_N^{\mathcal{N}}(\cdot;\mu)||_{\mathcal{Y}'} = ||\widehat{r}_N^{\mathcal{N}}(\mu)||_{\mathcal{Y}}$ with

$$\hat{r}_{N}^{\mathcal{N}}(\mu) = \sum_{q=1}^{Q_{f}} \theta_{f}^{(q)}(\mu) \, \hat{f}_{\mathcal{N}}^{q} + \sum_{i=1}^{N} u_{N}^{(i)}(\mu) \sum_{q=1}^{Q_{b}} \theta_{b}^{(q)}(\mu) \, \hat{b}_{\mathcal{N}}^{q,i}$$

can be computed using only N-dependent matrix-vector products and simple function evaluations for any reduced solution $u_N^{\mathcal{N}}(\mu)$. These offline/online-techniques are incorporated into the Greedy scheme in a straightforward manner.

3. Adaptive Reduced Basis Generation

In this section, we describe those issues that arise when avoiding fixed truth spaces $\mathcal{X}^{\mathcal{N}}$ and $\mathcal{Y}^{\mathcal{N}}$ and use adaptive methods instead. We assume that we have the following routine **SOLVE** at our disposal. In Appendix A below, we detail one possibility to realize **SOLVE** by an Adaptive Wavelet Galerkin Method (AWGM), but one could also use other schemes with the above properties such as adaptive finite element methods, see e.g. [18] for an overview.

SOLVE: $[\mathcal{A}, b, \varepsilon] \mapsto x^{\varepsilon}$: Approximation of $x = \mathcal{A}^{-1}b$ with $||x - x^{\varepsilon}||_{\mathcal{X}} \leq \varepsilon$ and in linear complexity.

3.1. Adaptive Snapshot Computation. We assume that we have an (adaptive) numerical solver SOLVE at our disposal that computes so-called ε -exact approximations $u^{\varepsilon}(\mu)$ of $u(\mu) \in \mathcal{X}$, i.e. that constructs parameter-dependent discrete spaces $\mathcal{X}^{\varepsilon}_{\mu}, \mathcal{Y}^{\varepsilon}_{\mu}$ of (possibly) arbitrary finite (from an RB point of view 'large', but in a certain sense minimal) dimension $\mathcal{N}(\mu, \varepsilon)$ with

$$(3.1) u^{\varepsilon}(\mu) \in \mathcal{X}^{\varepsilon}_{\mu}: b(u^{\varepsilon}(\mu), v; \mu) = f(v; \mu) \forall v \in \mathcal{Y}^{\varepsilon}_{\mu}$$

and

(3.2)
$$\|u(\mu) - u^{\varepsilon}(\mu)\|_{\mathcal{X}} \le \varepsilon.$$

The lack of common truth spaces for all parameters necessitates a re-interpretation of some RB ingredients which we will describe now. The reduced space is now spanned by *approximate* snapshots computed during the offline training phase, i.e.

(3.3)
$$X_N^{\varepsilon} := \operatorname{span}\{u^{\varepsilon}(\mu_i), i = 1, \dots, N\} =: \operatorname{span}\{\zeta_i^{\varepsilon}, i = 1, \dots, N\},$$

and the reduced solution $u_N^{\varepsilon}(\mu) \in X_N^{\varepsilon}$ is the Petrov-Galerkin projection onto this space (and the corresponding reduced inf-sup-stable test space Y_N^{ε}).

The adaptive setting now also allows us to consider the error with respect to the exact solution in \mathcal{X} , i.e.

(3.4)
$$e_N^{\varepsilon}(\mu) := u(\mu) - u_N^{\varepsilon}(\mu),$$

and not (only) the error w.r.t. a fixed and a priori given truth discretization. In fact, using standard arguments as above yields

(3.5)
$$\|e_N^{\varepsilon}(\mu)\|_{\mathcal{X}} \le \Delta_N^{\varepsilon}(\mu) = \frac{\|r_N^{\varepsilon}(\cdot;\mu)\|_{\mathcal{Y}'}}{\beta(\mu)} \le \frac{\gamma(\mu)}{\beta(\mu)} \|e_N^{\varepsilon}(\mu)\|_{\mathcal{X}},$$

with the residual defined as $r_N^{\varepsilon}(v;\mu) := f(v;\mu) - b(u_N^{\varepsilon}(\mu),v;\mu)$ for $v \in \mathcal{Y}$. This means that $\Delta_N^{\varepsilon}(\mu)$ is a surrogate for the true error $\|e_N^{\varepsilon}(\mu)\|_{\mathcal{X}}$. Note, however, that the computation of $\Delta_N^{\varepsilon}(\mu)$ requires the solution of an infinite-dimensional problem, namely the determination of the Riesz representation on \mathcal{Y} .

On the other hand, in this setting, the truth spaces $\mathcal{X}^{\varepsilon}_{\mu}$, $\mathcal{Y}^{\varepsilon}_{\mu}$ corresponding to $u^{\varepsilon}(\mu)$ are not known a priori, so that neither the set of involved local basis functions for representing $u^{\varepsilon}(\mu)$ (with parameter-dependent dimension $\mathcal{N}(\mu, \varepsilon)$) nor the Riesz representation of the corresponding residual $r^{\varepsilon}_{N}(\cdot;\mu)$ can be determined without computing $u^{\varepsilon}(\mu)$ itself.

Remark 3.1. The above formulated adaptive framework can also be interpreted as using different finite element meshes for different $\mu \in \mathcal{D}$ in the snapshot generation.

3.2. Adaptive Computation of the Error Estimator. Now we consider the computation of the error estimator which consists of the constant $\beta(\mu)$ and the dual norm of the residual. We aim at using adaptivity for approximating both. As for the residual, the norm of $r_N^{\varepsilon}(\cdot; \mu)$ would require computations on the infinite-dimensional space \mathcal{Y} . Hence, it seems natural also to use an adaptive method for computing the Riesz representions which are given in the infinite-dimensional setting as

(3.6) Find
$$\hat{r}_N^{\varepsilon}(\mu) \in \mathcal{Y}$$
: $(\hat{r}_N^{\varepsilon}(\mu), v)_{\mathcal{Y}} = r_N^{\varepsilon}(v; \mu) \quad \forall v \in \mathcal{Y}.$

This is the reason why we need to approximate Δ_N^{ε} by some $\Delta_{N,\delta}^{\varepsilon}$, where we need to choose the tolerance $\delta > 0$ appropriately. We start by analyzing in which sense

we need to approximate the residual or, slightly more general, an error estimator which is not completely computable.

Lemma 3.2. Assume $\Delta_N \geq 0$ is an equivalent bound for an error $e_N \geq 0$, i.e. there are absolute constants $0 < \underline{c}^e \leq \overline{C}^e < \infty$ with

(3.7)
$$\underline{c}^{\mathrm{e}}e_N \leq \Delta_N \leq \overline{C}^{\mathrm{e}}e_N.$$

Let $\delta \in [0,1)$ and $\Delta_{N,\delta}$ be an approximation of Δ_N satisfying

(3.8)
$$|\Delta_N - \Delta_{N,\delta}| \le \delta \cdot \begin{cases} \Delta_{N,\delta}, & \text{if } \Delta_N > 0, \\ 0, & \text{if } \Delta_N = 0. \end{cases}$$

Then, the approximate error estimator $\Delta_{N,\delta}$ is equivalent to the error in the sense

(3.9)
$$\frac{\underline{c}^{\mathrm{e}}}{1+\delta}e_{N} \leq \Delta_{N,\delta} \leq \frac{\overline{C}^{\mathrm{e}}}{1-\delta}e_{N}$$

Proof. If $\Delta_N = 0$, then $e_N = 0$ and $\Delta_N = \Delta_{N,\delta} = 0$, thus (3.9) holds. Now let $\Delta_N \neq 0$, then $(1 - \delta)\Delta_{N,\delta} \leq \Delta_N \leq \overline{C}^e e_N$, hence the upper inequality of (3.9). On the other hand, we have $(1 + \delta)\Delta_{N,\delta} \geq \Delta_N \geq \underline{c}^e e_N$, the lower bound in (3.9). \Box

Remark 3.3. (a) We apply Lemma 3.2 to $e_N = \|e_N^{\varepsilon}(\mu)\|_{\mathcal{X}} = \|u(\mu) - u_N^{\varepsilon}(\mu)\|_{\mathcal{X}}$ and $\Delta_N = \Delta_N^{\varepsilon}(\mu) = \frac{\|\hat{r}_N^{\varepsilon}(\mu)\|_{\mathcal{Y}}}{\beta(\mu)}$ so that (3.8) means

(3.10)
$$\left| \| \widehat{r}_{N}^{\varepsilon}(\mu) \|_{\mathcal{Y}} - \| \widehat{r}_{N,\delta}^{\varepsilon}(\mu) \|_{\mathcal{Y}} \right| \leq \delta \| \widehat{r}_{N,\delta}^{\varepsilon}(\mu) \|_{\mathcal{Y}}.$$

Obviously, (3.10) requires an adaptive method with a *relative* error bound.

(b) The right-hand side of the relative error criterion (3.8) only depends on the online computable quantity $\|\hat{r}_{N,\delta}^{\varepsilon}(\mu)\|_{\mathcal{Y}}$. Thus, (3.8) can be verified a posteriori for all $\mu \in \mathcal{D}$.

(c) The explicit knowledge of the equivalence constants yields the equivalence

(3.11)
$$\frac{1}{1+\delta} \|e_N^{\varepsilon}(\mu)\|_{\mathcal{X}} \le \Delta_{N,\delta}^{\varepsilon}(\mu) \le \frac{1}{1-\delta} \frac{\gamma(\mu)}{\beta(\mu)} \|e_N^{\varepsilon}(\mu)\|_{\mathcal{X}},$$

i.e., also the computable error $\Delta_{N,\delta}^{\varepsilon}(\mu)$ can be used as a surrogate for the true error $\|e_N^{\varepsilon}(\mu)\|_{\mathcal{X}}$. Of course, we have to take into account that the amount of work required to compute $\Delta_{N,\delta}^{\varepsilon}(\mu)$ grows as $\delta \to 0$. Finally, in the Greedy training phase, the equivalence of an error estimator is more important than its rigor, as the main requirement on the surrogate $\Delta_{N,\delta}^{\varepsilon}(\mu)$ is a correct choice of the next snapshot parameter in terms of the maximization.

We have seen that **SOLVE** provides us with an adaptive scheme for an absolute error. It is not hard to see how to use it to derive a relative error tolerance.

Lemma 3.4. Let a > 0 and $(b_{\rho})_{\rho > 0}$ be a sequence with $|a - b_{\rho}| \leq \rho$. Moreover, let $\delta \in (0,1)$ be given. Then, choosing $\rho^*(\delta) < \delta \frac{a}{2}$ yields $|a - b_{\rho^*(\delta)}| \leq \delta |b_{\rho^*(\delta)}|$.

Proof. Since $\delta < 1$ we have $|a - b_{\rho^*(\delta)}| < \frac{a}{2}$, thus $b_{\rho^*(\delta)} \in (\frac{a}{2}, \frac{3a}{2})$, in particular $b_{\rho^*(\delta)} > \frac{a}{2} > 0$. Then, $|a - b_{\rho^*(\delta)}| \le \rho^*(\delta) < \delta \frac{a}{2} < \delta b_{\rho^*(\delta)} = \delta |b_{\rho^*(\delta)}|$, which proves the claim.

Note, however, that in the RB setting, a is usually unknown, so that $\rho^*(\delta)$ cannot be determined a priorily.

3.3. Adaptive Greedy Algorithm. Now we have all ingredients at hand to formulate a fully adaptive version of the Greedy algorithm in Algorithm 2. The adaptive computations take place in line 2 concerning the error estimator and in line 5 for the snapshot. If Algorithm 2 terminates, we get $\Delta_{N,\delta}^{\varepsilon}(\mu) < \widetilde{\text{tol}}$ for all $\mu \in \Xi_{\text{train}}$, hence

$$\max_{\mu \in \Xi_{\text{train}}} \|e_N^{\varepsilon}(\mu)\|_{\mathcal{X}} \le (1+\delta) \widetilde{\text{tol}}.$$

Fixing some $\delta \in (0, 1)$, one has to use $\widetilde{\text{tol}} = \frac{\text{tol}}{1+\delta}$ in line 3 in order to reach a desired tolerance tol for the true error on Ξ_{train} . (Note that as usual, the quality of Ξ_{train} is important in order to control the error for all $\mu \in \mathcal{D}$.)

| $\overline{\textbf{Algorithm 2} [X_{N,\delta}^{\varepsilon}]} = \textbf{AdaptGreedy}[\widetilde{\text{tol}}, N_{\max}, \varepsilon, \delta, \Xi_{\text{train}}]$ |
|--|
| 1: for $N = 0, 1,, N_{\max}$ do |
| 2: Choose $\mu^* := \operatorname{argmax}_{\mu \in \Xi_{\operatorname{train}}} \Delta_{N,\delta}^{\varepsilon}(\mu).$ |
| 3: if $\Delta_{N,\delta}^{\varepsilon}(\mu^*) < \widetilde{\mathrm{tol}}$ then return |
| 4: $N \leftarrow N + 1$. |
| 5: Compute snapshot $u^{\varepsilon}(\mu^*)$. |
| 6: Update reduced basis: $X_{N\delta}^{\varepsilon} = X_{N-1\delta}^{\varepsilon} \cup \{u^{\varepsilon}(\mu^*)\}.$ |
| 7: end for |
| |

It seems natural that the tolerance $\varepsilon > 0$ bounds the reduction error $e_N^{\varepsilon}(\mu) = u(\mu) - u_N^{\varepsilon}(\mu)$ from below in the sense that the error cannot be smaller than the accuracy of the snapshot approximations. A result from [3] shows that this can lead to a stalling of the Greedy training at a certain level. As usual, the benchmark for the Greedy algorithm is the *Kolmogorov n-width* for some $\Sigma \subset \mathcal{X}$, i.e.,

$$d_n(\Sigma) := \inf_{\dim(\mathcal{X}_n)=n} \sup_{f \in \Sigma} \min_{g \in \mathcal{X}_n} \|f - g\|_{\mathcal{X}}.$$

Theorem 3.5 ([3]). Let $\mathcal{M}(\mu) := \{u(\mu) : \mu \in \mathcal{D}\}$ be compact and suppose that $d_0(\mathcal{M}(\mu)) \leq M, d_n(\mathcal{M}(\mu)) \leq Mn^{-\theta}$ for some $M, \theta > 0$. Then, the approximation $X_{N,\delta}^{\varepsilon} = \mathbf{AdaptGreedy}[tol, N_{\max}, \varepsilon, \delta, \mathcal{D}]$ satisfies

$$\sup_{\sigma \in \mathcal{M}(\mu)} \min_{g \in X_{N,\delta}^{\varepsilon}} \|\sigma - g\|_{\mathcal{X}} \le C \max\{Mn^{-\theta}, \varepsilon\}, \qquad \gamma := \frac{1-\delta}{1+\delta} \frac{\underline{c}^{\mathrm{e}}}{\overline{c}^{\mathrm{e}}}$$

with a constant $C = C(\theta, \gamma)$.

This result tells us that the RB Greedy training converges quasi-optimally compared to the Kolmogorov *n*-width until an ε -dependent error level is reached. Even though this is a worst case result, we have observed in several numerical tests that **AdaptGreedy** indeed reaches a terminal stage (a "plateau" related to $C\varepsilon$). This happens when in the snapshot selection in line 2 a previously chosen parameter is re-selected. As we will discuss below, this is not an effect of inaccurate numerical computations, but inherently linked to the adaptive setting and can be improved by appropriate *snapshot accuracy strategies*.

3.4. (Non-)Reproduction of Snapshots. As we have pointed out in Remark 2.3, on a fixed truth discretization we have that $\Delta_N^{\mathcal{N}}(\mu) = 0$ (up to numerical influences) for all $\mu \in S_N$, i.e. the error bound vanishes on the set of snapshot parameters, as all snapshots $u^{\mathcal{N}}(\mu) \in \mathcal{X}^{\mathcal{N}}$, $\mu \in S_N$, can be reconstructed exactly from the basis functions and the Riesz representation for the error estimator is only based upon $\mathcal{X}^{\mathcal{N}}$, $\mathcal{Y}^{\mathcal{N}}$. As we will explain now this is not the case in the adaptive framework. The reason is that $u^{\varepsilon}(\mu) \in \mathcal{X}^{\varepsilon}_{\mu}$ but $u^{\varepsilon}_{N}(\mu) \notin \mathcal{X}^{\varepsilon}_{\mu}$. In fact, we only have

(3.12)
$$u_N^{\varepsilon}(\mu) \in \bigcup_{\tilde{\mu} \in S_N} \mathcal{X}_{\tilde{\mu}}^{\varepsilon} =: \mathcal{X}^{\varepsilon, S_N},$$

and similarly $Y_N^{\varepsilon} \not\subseteq Y_{\mu}^{\varepsilon}$. Hence, the argument using Petrov-Galerkin orthogonality as in Remark 2.3 fails. In fact, note that $e_N^{\varepsilon}(\mu) = u(\mu) - u_N^{\varepsilon}(\mu)$ is the error w.r.t. the unknown solution $u(\mu)$, whereas $e_N^{\mathcal{N}}(\mu) = u^{\mathcal{N}}(\mu) - u_N^{\mathcal{N}}(\mu)$ involves the 'truth' solution, which is in principle computable (up to numerical precision). This is important since in the 'classical' case $u^{\mathcal{N}}(\mu)$ is used as a snapshot, whereas in the adaptive setting $u(\mu)$ cannot be computed and has to be replaced by an approximation $u^{\varepsilon}(\mu)$. Hence, $b(u^{\varepsilon}(\mu) - u_N^{\varepsilon}(\mu), v_N^{\varepsilon}; \mu)$ will in general *not* vanish! This means that –as opposed to the 'classical' RBM– snapshots are *not reproduced* in the adaptive setting. Reproduction of RB basis functions is *not* a consequence of the fact that the RB spaces are spanned by those functions, but a consequence of the Petrov-Galerkin orthogonality.

Of course, one could use $\mathcal{X}^{\varepsilon,S_N}$ defined in (3.12) as a joint common truth space. However, if the discretizations for different μ are significantly different, this would be by far too costly, in particular because already computed snapshots would have to be updated to the new truth space in each iteration.

Hence, we face a reproduction error, which is now investigated a little further.

Proposition 3.6. Let $b(\cdot, \cdot; \mu) : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ be inf-sup stable on $X_N^{\varepsilon} \times Y_N^{\varepsilon}$ with inf-sup constant $\beta_N(\mu)$. Then for all $\mu_i \in S_N$, we have

(3.13)
$$\max\left\{\|u(\mu_i) - u_N^{\varepsilon}(\mu_i)\|_{\mathcal{X}}, \|u^{\varepsilon}(\mu_i) - u_N^{\varepsilon}(\mu_i)\|_{\mathcal{X}}\right\} \le \frac{\gamma(\mu_i)}{\beta_N(\mu_i)}\varepsilon(\mu_i),$$

where $\varepsilon(\mu_i)$ denotes the accuracy of snapshot $u^{\varepsilon}(\mu_i)$.

Proof. Let $\mu_i \in S_N$. As $X_N^{\varepsilon} \subset \mathcal{X}$, $Y_N^{\varepsilon} \subset \mathcal{Y}$, we have Petrov-Galerkin orthogonality w.r.t. the exact solution, i.e. $b(u(\mu) - u_N^{\varepsilon}(\mu), v_N; \mu) = 0$ for all $v_N \in Y_N^{\varepsilon}$. This implies the quasi-best approximation property ([18], cf. Céa's Lemma) $||u(\mu) - u_N^{\varepsilon}(\mu)||_{\mathcal{X}} \leq \frac{\gamma(\mu)}{\beta_N(\mu)} \inf_{w_N \in X_N^{\varepsilon}} ||u(\mu) - w_N||_{\mathcal{X}}$. As snapshots $u^{\varepsilon}(\mu_i)$, $\mu_i \in S_N$ are in X_N^{ε} , the first inequality follows with $\inf_{w_N \in X_N^{\varepsilon}} ||u(\mu) - w_N||_{\mathcal{X}} \leq \varepsilon(\mu_i)$. Moreover, we have

$$\beta_{N}(\mu_{i}) \leq \inf_{u_{N}\in X_{N}^{\varepsilon}} \sup_{v_{N}\in Y_{N}^{\varepsilon}} \frac{b(u_{N}, v_{N}; \mu_{i})}{\|u_{N}\|_{\mathcal{X}} \|v_{N}\|_{\mathcal{Y}}} \leq \sup_{v_{N}\in Y_{N}^{\varepsilon}} \frac{b(u^{\varepsilon}(\mu_{i}) - u_{N}^{\varepsilon}(\mu_{i}) \pm u(\mu_{i}), v_{N}; \mu_{i})}{\|u^{\varepsilon}(\mu_{i}) - u_{N}^{\varepsilon}(\mu_{i})\|_{\mathcal{X}} \|v_{N}\|_{\mathcal{Y}}}$$
$$= \sup_{v_{N}\in Y_{N}^{\varepsilon}} \frac{b(u^{\varepsilon}(\mu_{i}) - u(\mu_{i}), v_{N}; \mu_{i})}{\|u^{\varepsilon}(\mu_{i}) - u_{N}^{\varepsilon}(\mu_{i})\|_{\mathcal{X}} \|v_{N}\|_{\mathcal{Y}}} \leq \frac{\gamma(\mu_{i})\varepsilon(\mu_{i})}{\|u^{\varepsilon}(\mu_{i}) - u_{N}^{\varepsilon}(\mu_{i})\|_{\mathcal{X}}}.$$

As a consequence, it may happen that $\mu^* := \arg \max_{\tilde{\mu} \in \Xi_{\text{train}}} \Delta_{N+1,\delta}^{\varepsilon}(\tilde{\mu}) \in S_N$ if $\varepsilon \frac{\gamma(\mu^*)}{\alpha(\mu^*)} \ge \max_{\tilde{\mu} \in \Xi_{\text{train}} \setminus S_N} \Delta_{N+1,\delta}^{\varepsilon}(\tilde{\mu})$. The first idea is to replace line 2 in Algorithm

2 by

(3.14) Choose
$$\mu^* := \operatorname{argmax}_{\mu \in \Xi_{\operatorname{train}} \setminus S_N} \Delta_{N,\delta}^{\varepsilon}(\mu)$$

i.e. to remove already chosen snapshot parameters from the training set Ξ_{train} .

Theorem 3.5 suggests equal tolerances ε for all snapshots, as the error plateau is determined by the worst case estimate. However, the parameter-dependent scaling factor $\frac{\gamma(\mu_i)}{\beta_N(\mu_i)}$ in Prop. 3.6 indicates that μ -dependent accuracies might be more appropriate. Another idea is therefore to start with a common tolerance $\varepsilon(\mu) \equiv \varepsilon^{(0)}$ and to adjust this if a repeated selection of $\mu^* \in S_N$ signals that the bound in (3.13) is too large. This leads us to the following snapshot refinement strategy:

(SR) Given $\varepsilon^{(0)}$, set for some pre-specified reduction factor $\varrho_u \in (0,1)$

(3.15)
$$\varepsilon^{(n+1)} := \begin{cases} \varrho_u \cdot \varepsilon^{(n)} & \text{if } \mu^* \in S_n \\ \varepsilon^{(n)} & \text{else.} \end{cases}$$

3.5. Adaptive Offline-online Decomposition. As in the 'standard' RBM, we assume an affine decomposition of $b(\cdot, \cdot; \mu)$ and $f(\cdot; \mu)$ as in (2.11).

3.5.1. Computation of the RB Solution. It is readily seen that $u_N^{\varepsilon}(\mu)$ can be computed online-efficient like in the standard case. In fact, the matrix and right-hand side terms $\mathbf{B}_{\varepsilon,N}^{(q)} := \left[b^{(q)}(u^{\varepsilon}(\mu_i), u^{\varepsilon}(\mu_j))\right]_{i,j=1,\dots,N}, q = 1,\dots,Q_b$, and $\mathbf{F}_{\varepsilon,N}^{(q)} := \left[f^{(q)}(u^{\varepsilon}(\mu_j))\right]_{j=1,\dots,N}$ can be computed offline. Then, $\mathbf{B}_{\varepsilon,N}(\mu)$ and $\mathbf{F}_{\varepsilon,N}(\mu)$ are formed as in the classical case and the solution $\mathbf{u}_{\varepsilon,N}(\mu) = [u_N^{\varepsilon,n}(\mu)]_{n=1,\dots,N}$ of the linear system corresponding to (2.12) yields the RB solution $u_N^{\varepsilon}(\mu)$.

3.5.2. Error Estimator. In principle we could follow the same path as above and solve the analogue to (2.13), i.e.,

(3.16a) Find
$$\hat{b}^{q,n} \in \mathcal{Y}$$
: $(\hat{b}^{q,n}, v)_{\mathcal{Y}} = -b^{(q)}(\zeta_n^{\varepsilon}, v) \quad \forall v \in \mathcal{Y},$

(3.16b) Find
$$\hat{f}^q \in \mathcal{Y}$$
: $(\hat{f}^q, v)_{\mathcal{Y}} = f^{(q)}(v) \quad \forall v \in \mathcal{Y}.$

Note that (3.16a) and (3.16b) are again variational problems on the infinite-dimensional space \mathcal{Y} . Using corresponding adaptive schemes yield (for given tolerancs $\delta_{b^{q,n}}, \delta_{f^q} > 0$) approximations $\hat{b}^{q,n}_{\delta}, \hat{f}^q_{\delta}$ satisfying

$$\|\widehat{b}^{q,n} - \widehat{b}^{q,n}_{\delta}\|_{\mathcal{Y}} < \delta_{b^{q,n}}, \qquad \|\widehat{f}^q - \widehat{f}^q_{\delta}\|_{\mathcal{Y}} < \delta_{f^q}.$$

As in §2.4, we could compute the required inner products and would obtain an approximation, say $\Delta_{N,\delta}^{\varepsilon, \text{aff}}(\mu)$ of $\Delta_N^{\varepsilon}(\mu)$. However, we need to verify the relative accuracy in (3.8). This can be done as follows. Denoting by δ_{f^q} , $\delta_{b^{q,n}}$ the accuracies of \hat{f}^q_{δ} and $\hat{b}^{q,n}_{\delta}$, respectively, it holds that

$$\begin{aligned} \|\hat{r}_{N}(\mu) - \hat{r}_{N,\delta}^{\mathrm{aff}}(\mu)\|_{\mathcal{Y}}^{2} &= \left\|\sum_{q=1}^{Q_{f}} \theta_{f}^{(q)}(\mu) \left[\hat{f}^{q} - \hat{f}_{\delta}^{q}\right] + \sum_{n=1}^{N} u_{N}^{\varepsilon,n}(\mu) \sum_{q=1}^{Q_{b}} \theta_{b}^{(q)}(\mu) \left[\hat{b}^{q,n} - \hat{b}_{\delta}^{q,n}\right]\right\|_{\mathcal{Y}}^{2} \\ &\leq \sum_{q,q'=1}^{Q_{f}} \left(\theta_{f}^{(q)}(\mu) \theta_{f}^{(q')}(\mu)\right)^{+} \delta_{f^{q}} \delta_{f^{q'}} + 2 \sum_{n=1}^{N} \sum_{q=1}^{Q_{f}} \sum_{q'=1}^{Q_{b}} \left(u_{N}^{\varepsilon,n}(\mu) \theta_{f}^{(q)}(\mu) \theta_{b}^{(q')}(\mu)\right)^{+} \delta_{f^{q}} \delta_{b^{q',n}} \\ &+ \sum_{n,n'=1}^{N} \sum_{q,q'=1}^{Q_{b}} \left(u_{N}^{\varepsilon,n}(\mu) u_{N}^{\varepsilon,n'}(\mu) \theta_{b}^{(q)}(\mu) \theta_{b}^{(q)}(\mu)\right)^{+} \delta_{b^{q,n}} \delta_{b^{q',n'}} =: \delta_{\mathrm{aff}}^{\mathrm{r}}(\mu)^{2}. \end{aligned}$$

If then $\delta_{\mathrm{aff}}^{\mathrm{r}}(\mu) < \|\hat{r}_{N,\delta}^{\mathrm{aff}}(\mu)\|_{\mathcal{V}}$, criterion (3.8) is fulfilled. Note that we can arrange $\delta_{\mathrm{aff}}^{\mathrm{r}}(\mu) > 0$ to be as small as we wish by using sufficiently small tolerances δ_{f^q} , $\delta_{b^{q,n}}$ — at the expense of possibly high numerical cost. It is however not possible to do this completely offline, since $\delta_{\mathrm{aff}}^{\mathrm{r}}(\mu)$ is μ -dependent. This means that we possibly need to update the adaptive approximations during the online-phase. Thus, we verify the validity of (3.8) a posteriori as follows: Since we can compute $\delta_{\mathrm{aff}}^{\mathrm{r}}(\mu)$, we check if $\delta_{\mathrm{aff}}^{\mathrm{r}}(\mu) \leq \rho \|\hat{r}_{N,\delta}^{\mathrm{aff}}(\mu)\|_{\mathcal{V}}$ for a given $\rho \in (0, 1)$. If not, we decrease the involved tolerances. Due to the convergence of the used adaptive schemes, this approach in fact converges and results in an approximate estimate $\Delta_{N,\delta}^{\varepsilon}(\mu)$ which is equivalent to $\|e_{N}^{\varepsilon}(\mu)\|_{\mathcal{X}}$ as in (3.11). However, the above estimate of $\|\hat{r}_{N}(\mu) - \hat{r}_{N,\delta}^{\mathrm{aff}}(\mu)\|_{\mathcal{V}}$ in terms of $\delta_{\mathrm{aff}}^{\mathrm{r}}(\mu)$ is rather crude so that we expect non-optimal numerical performance — as we shall also see in our numerical experiments below. Yet, if we store the adaptive approximation (i.e., the 'active' wavelet coefficients) of each Riesz representation, we only need to update those discretizations if the choice of μ requires additional accuracy.

4. Numerical Experiments

4.1. **Data for the Experiments.** As numerical examples, we consider an elliptic problem as well as a parabolic (time-periodic) one in space-time formulation. Both settings are described below in detail.

4.1.1. Elliptic Equation. We consider heat conduction in a 2D thermal block $\Omega = (0, 1)^2$ consisting of two subdomains $\Omega_0 = [0.5, 1] \times [0, 1]$, $\Omega_1 = [0, 0.5] \times [0, 1]$, with different conductivities $\mu_0 = 1$, $\mu_1 \in [0.01, 100]$, [19]. The heat influx is modeled as a constant local source on different parts $\tilde{\Omega}_i$, $i = 1, \ldots, 9$, of the domain, where the current location depends on a (discrete) parameter $\mu_2 \in \{1, \ldots, 9\}$, see Figure 1a. We impose homogeneous Dirichlet boundary conditions on $\Gamma_D := \partial\Omega \cap \{x = 0 \lor x = 1\}$ and homogeneous Neumann conditions on $\Gamma_N := \partial\Omega \cap \{y = 0 \lor y = 1\}$. The variational formulation then reads: Find $u \in \mathcal{X} := H_D^1(\Omega) = \{v \in H^1(\Omega) : v = 0 \text{ on } \Gamma_D\}$ such that

$$\int_{\Omega_0} \nabla u \cdot \nabla v + \mu_1 \int_{\Omega_1} \nabla u \cdot \nabla v = (f(\mu_2), v)_{L_2(\Omega)} \ \forall v \in \mathcal{X}, \quad f(\mu_2) := \sum_{i=1}^9 \delta_{\mu_2, i} \mathbb{1}_{\{\widetilde{\Omega}_i\}}.$$



(A) Thermal block with 9 local sources.

(B) Basis functions for $\mu_1 = 0.01$, $\mu_2 = 8$ (left) and $\mu_1 = 0.01$, $\mu_2 = 5$ (right).

FIGURE 1. Domain of the thermal block example and selected (non-orthogonalized) snapshots.

We employ a multitree-based AWGM (see Appendix A below) with a tensor basis consisting of bi-orthogonal B-spline wavelets from [9] of order $d_x = m_x = 2^d$ and $L_2(0, 1)$ -orthonormal (multi-)wavelets as in [20] of order $d_y = m_y = 2$, with homogeneous boundary conditions. In Figure 1b, we display snapshots corresponding to two different parameter values. It is obvious that an adaptive discretization is useful in this example.

4.1.2. *Parabolic Periodic Space-Time Equation*. As a second example, we consider the time-periodic convection-diffusion-reaction (CDR) equation

$$\begin{cases} u_t - u_{xx} + \mu_1 \beta(x) u_x + \mu_2 u = \cos(2\pi t) & \text{on } \Omega = (0, 1), \\ u(t, 0) = u(t, 1) & \text{for all } t \in [0, T], \\ u(0, x) = u(T, x) = 0 & \text{on } \overline{\Omega}, \end{cases}$$

with coefficient function $\beta(x) = 0.5 - x$. Setting $V := H_0^1(\Omega), H_{\text{per}}^1(0,T) := \{v \in H^1(0,T) : v(0) = v(T)\}$, we define the spaces $\mathcal{Y} := L_2(0,T;V)$ and $\mathcal{X} := L_2(0,T;V) \cap H_{\text{per}}^1(0,T;V')$, i.e.,

(4.1)
$$\mathcal{X} = \{ v \in L_2(0,T;V) : v_t \in L_2(0,T;V'), \ v(0) = v(T) \text{ in } H \},\$$

where \mathcal{X} is equipped with the norm $\|v\|_{\mathcal{X}}^2 := \|v\|_{L_2(0,T;V)}^2 + \|v_t\|_{L_2(0,T;V')}^2$, $v \in \mathcal{X}$. Note that v(0), v(T) are well-defined due to $H^1(0,T) \subset C([0,T])$ and $\{v \in L_2(0,T;V) : v_t \in L_2(0,T;V')\} \subset C(0,T;H)$, e.g. [8]. We obtain the variational problem:

(4.2) Find
$$u \in \mathcal{X}$$
: $b(u, v; \mu) = f(v)$ $\forall v \in \mathcal{Y}, \mu = (\mu_1, \mu_2)$

with forms $b(\cdot, \cdot; \mu) : \mathcal{X} \times \mathcal{Y} \times \mathcal{D} \to \mathbb{R}, f(\cdot) : \mathcal{Y} \to \mathbb{R}$ given by

(4.3)
$$b(u,v;\mu) := \int_0^T [\langle v(t), u_t(t) \rangle_{V \times V'} + a(u(t), v(t);\mu)] dt$$

 $f(v) := \int_0^T \cos(2\pi t) \langle v(t), 1 \rangle_{V \times V'} dt \text{ and } a(\phi, \eta; \mu) = (\phi_x, \eta_x)_{L_2(\Omega)} + \mu_1(\beta \phi_x, \eta)_{L_2(\Omega)} + \mu_2(\phi, \eta)_{L_2(\Omega)}.$

As bases we use space-time tensor functions: In time, we use a collection of biorthogonal B-spline wavelets on \mathbb{R} of order $d_t = m_t = 2$, periodized onto [0, T], [23]. The spatial basis is chosen as bi-orthogonal B-spline wavelets of order $d_x = m_x = 2$ with homogeneous boundary conditions from [9]. The test basis is a tensor product of the above mentioned linear B-spline wavelets with 2 vanishing moments from [9] with homogeneous boundary conditions in the univariate spatial basis.

In this example, the snapshots have different temporal evolutions. Since time is a 'normal' variable in a space-time variational formulation, this means that different discretizations for the snapshots in space-time pay off. This justifies adaptivity.

4.2. Strong Adaptive Greedy Training and Snapshot Strategies. We start by investigating the snapshot selection in a Greedy training. In order to distinguish the effects of approximative snapshots from those of approximative error estimators, we first consider only *strong* Greedy methods, i.e. using the error $e_N^{\varepsilon}(\mu)$ instead of the estimate $\Delta_N^{\varepsilon}(\mu)$ (or its approximation $\Delta_{N,\delta}^{\varepsilon}(\mu)$) for the snapshot selection. This is done by computing a sufficiently accurate approximation to each $u(\mu)$. We compare a standard training with different snapshot accuracies, denoted by (SF),

^dThe meaning of the parameters d and m are explained in Appendix A below.

with the search (Sw/o) over the restricted parameter set as in (3.14) as well as the snapshot refinement strategy (SR) from (3.15), using $\rho_u = 0.1$. The results for both examples are shown in Figure 2. As we see, there are plateaus due to the multiple selection of snapshots as described in §3.4 above and these plateaus appear later for increased snapshot accuracies, as expected. Moreover, we see that the strategy (Sw/o) can avoid the plateaus but does not admit the quasi-optimal convergence rate of the Greedy training. The update (SR) proves very effective in the CDR example while reducing the total computational cost in comparison to (SF) with fixed small tolerance $\varepsilon = 0.005$.



FIGURE 2. Strong Greedy error in adaptive RB computation.

4.3. Weak Adaptive Greedy Training and Error Estimator Strategies. To investigate the performance of the approximate error estimator $\Delta_{N,\delta}^{\varepsilon}(\mu)$, we compare the strong Greedy benchmark from §4.2 with weak Greedy trainings. Here, we compute the error estimator $\Delta_{N,\delta}^{\varepsilon}(\mu)$ directly, i.e. via adaptive solutions of (3.6) for each parameter, denoted by (ED). Further, we use the affine decomposition from §3.5.2, testing if $\delta_{\text{aff}}^{r}(\mu)$ is small enough for each $\mu \in \Xi_{\text{train}}$, called (EA). Note that both estimators can be (and are) scaled by $1 + \delta$ to be rigorous (we use $\delta = 0.99$ to minimize computational cost). However, the direct computation is of course not online efficient, while the affine version suffers from possibly large offline cost. The results are shown in Figures 3, 4. We see in Fig. 3a, 4a that the effectives of (ED) are very good: less than 10 in the case of the CDR example and almost exact for the thermal block. The affine bounds (EA) perform less well and show a significantly weaker convergence (CDR example) or a large overestimation factor (thermal block).



FIGURE 3. Thermal block – Weak Greedy training, rigorous bounds.



FIGURE 4. CDR example – Weak Greedy training, rigorous bounds.

That the poorer performance of (EA) is only an issue of the error bound sharpness is shown in Figures 3b, 4b. Here, we compute the "true" maximum error $\max_{\mu \in \Xi_{\text{train}}} \|e_N^{\varepsilon}(\mu)\|$ with respect to the reference solutions as an indicator of the real RB basis quality. It is obvious that the constructed bases are very near the benchmark of the strong Greedy scheme. This is equally true for the bases constructed using (EA) – here, even though the error estimate is worse, the *same snapshot parameters* are selected (in case of the thermal block in a slightly different order), so that the bases obtained from (ED) and (EA) do not or almost not differ.

Additionally, we consider for the second example two heuristic estimators that are both based on the affine decomposition (3.16). For the first, denoted by (HF), we fix the accuracies $\delta_{b^{q,n}}, \delta_{f^q} > 0$ in the adaptive computations for the Risz representations and combine them to an error indicator without checking if $\delta^{\rm r}_{\rm aff}(\mu)$ is below the desired tolerance, so that equivalence and rigor are not guaranteed. The second, called (HR), mirrors the snapshot update strategy (SR): Starting with $\delta^{(0)}_{b^{q,n}} = \delta^{(0)}_{f^q}$, we reduce both tolerances by factors ϱ_f , $\varrho_b \in (0,1)$ if a parameter is re-chosen, i.e.

$$\delta_{f^q}^{(n+1)} := \begin{cases} \varrho_f \cdot \delta_{f^q}^{(n)} & \text{if } \mu^* \in S_n, \\ \delta_{f^q}^{(n)} & \text{else,} \end{cases} \qquad \delta_{b^{q,n}}^{(n+1)} := \begin{cases} \varrho_b \cdot \delta_{b^{q,n}}^{(n)} & \text{if } \mu^* \in S_n, \\ \delta_{b^{q,n}}^{(n)} & \text{else,} \end{cases}$$

using $\rho_f = \rho_b = 0.1$. The results are displayed in Figure 5.



FIGURE 5. CDR example – Weak Greedy training, heuristic bounds.

We see that the efficiency of the heuristic error estimators is close to that of the rigorous and equivalent bound (ED) and significantly better than the certified equivalent affine bound (EA), Fig. 5a. The heuristic indicator (HF) mirrors the errors in the different snapshot scenarios correctly: for a refined snapshot (SR) the error indicator is smaller than in scenario (SF). However, the snapshot selection is less optimal: the same parameter ($\mu = (0, -9)$) is selected repeatedly, which is reflected in the almost identical "true" error of both bases depicted in Fig. 5b. The update (HR) is a remedy in this case: using a more exact error indicator $\Delta_{N,\delta}^{\varepsilon}(\mu)$ avoids the termination of the Greedy scheme as it selects a different parameter. The so constructed basis is even comparable to the strong Greedy benchmark (which includes only snapshots with original accuracy ε_0).

| N | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---------------------------------------|-------|-------|-------|--------|--------|--------|--------|
| (ED) (av. over Ξ_{train}) | 175 | 183 | 200 | 251 | 509 | 684 | 1611 |
| (EA) | 2335 | 76876 | 85100 | 106997 | 517771 | 667823 | _ |
| (HF) | 41127 | 59705 | 62648 | 74459 | 74529 | 76082 | _ |
| (HR) | 52604 | 70010 | 72812 | 84409 | 521927 | 656590 | 956996 |

TABLE 1. Example 2 – Number of active wavelets for the computation of the Riesz representations $\hat{r}_{N,\delta}(\mu)$ in scenario (SR).

It thus appears that the heuristic estimators can prove valid substitutes for the certified equivalent estimators (ED), (EA). In Table 1, we compare the respective offline computational work needed for the construction of $\Delta_{N,\delta}^{\varepsilon}(\mu)$ in the CDR example, measured by the (total) size of the (parameter-independent) local adaptive wavelet bases constructed during the computations of the Riesz representors. For the parameter-dependent bases of the direct estimate (ED), we indicate the average size over $\mu \in \Xi_{\text{train}}$. The large discrepancy between the average parameter-dependent representor size for (ED) and that for the union of the representor components in the affine decompositions underlines the twofold problem that the estimate $\delta_{\text{aff}}^{\text{r}}(\mu)$ is rather crude and that additionally the representor accuracy is oriented on the worst parameter case, so that the individual discretizations are extremely (and overly) fine. This effectively leads to even offline infeasible cost for the equivalent estimator (EA), while it is large but manageable for (HR).

Obviously, there is still room and need for improvements. We will investigate this further in the near future.

APPENDIX A. ADAPTIVE WAVELET GALERKIN METHODS (AWGMS)

To obtain an adaptive approximation for the snapshots $u^{\mathcal{N}}(\mu)$ as well as the error estimators $\Delta_{N}^{\mathcal{N}}(\mu)$ we employ adaptive wavelet Galerkin methods (AWGMs) that have first been introduced in [6, 7] for stationary problems and extended to space-time variational parabolic problems in [21]. Here, we used multitree-based versions developed in [13, 14, 15, 16], which we briefly review. Let $\mathcal{A} : \mathcal{X} \to \mathcal{Y}'$ be a linear differential (or integral) operator which may or may not depend on $\mu \in \mathcal{D}$. Given some $b \in \mathcal{Y}'$, we look for $x \in \mathcal{X}$ such that

(A.1)
$$\mathcal{A}x = b \quad \text{in } \mathcal{Y}'.$$

A.1. Equivalent Bi-infinite Matrix-Vector Problem. Variational equations of the form (A.1) can be reformulated as *equivalent* ℓ_2 -problems by considering *Riesz bases* of the Hilbert spaces \mathcal{X}, \mathcal{Y} . We call $\Upsilon := \{\gamma_i : i \in \mathbb{N}\} \subset \mathbb{Z}$ a *Riesz basis*

for a separable Hilbert space \mathcal{Z} if its linear span is dense in \mathcal{Z} and if there exist c, C > 0 such that

(A.2)
$$c \|\mathbf{v}\|_{\ell_2(\mathbb{N})}^2 \le \|v\|_{\mathcal{Z}}^2 \le C \|\mathbf{v}\|_{\ell_2(\mathbb{N})}^2 \quad \forall \mathbf{v} = (v_i)_{i \in \mathbb{N}} \in \ell_2(\mathbb{N}), \ v = \sum_{i=1}^{\infty} v_i \gamma_i.$$

For \mathcal{X} , \mathcal{Y} , we denote these Riesz wavelet bases by

(A.3)
$$\widehat{\Psi}^{\mathcal{X}} := \left\{ \widehat{\psi}^{\mathcal{X}}_{\lambda} : \lambda \in \widehat{\mathcal{J}} \right\} \subset \mathcal{X}, \qquad \widecheck{\Psi}^{\mathcal{Y}} := \left\{ \widecheck{\psi}^{\mathcal{Y}}_{\lambda} : \lambda \in \widecheck{\mathcal{J}} \right\} \subset \mathcal{Y},$$

for countable index sets $\hat{\mathcal{J}}$, $\check{\mathcal{J}}$. Such bases can be constructed by first building univariate wavelet bases $\Psi = \{\psi_{\lambda} : \lambda \in \mathcal{J}\}$ for $L_2(0, 1)$ that are sufficiently smooth to constitute (after a proper normalization) also Riesz bases for a whole range of Sobolev spaces $H^s(0, 1)$, $s \in (-\tilde{\gamma}, \gamma)$, where $\gamma, \tilde{\gamma} > 0$ depend on the choice of the wavelets, cf. [23]. Typically the index takes the form $\lambda = (j, k)$, where $|\lambda| := j$ denotes the *level* (e.g. | supp $\psi_{\lambda}| \sim 2^{-|\lambda|}$) and k the location in (0, 1), e.g. the center of its support. We consider piecewise polynomial wavelets of order d (degree plus one). Wavelets are oscillating ("small waves") which is reflected by their degree m of vanishing moments, i.e., $\int_0^1 x^r \psi_{\lambda}(x) dx = 0$ for all $|\lambda| > 0$ and all $0 \le r \le m - 1$, where $|\lambda| = 0$ denotes the coarsest level, $0 = \min_{\lambda \in \mathcal{J}} |\lambda|$. Those functions are no 'true' wavelets but e.g. splines (scaling functions). The above mentioned constants γ and $\tilde{\gamma}$ are determined by d, m and \tilde{d} , \tilde{m} , which are the corresponding parameters of the *dual* wavelet basis $\tilde{\psi} = \{\tilde{\psi}_{\lambda} : \lambda \in \mathcal{J}\}$ with $\int_0^1 \psi_{\lambda}(x) \tilde{\psi}_{\lambda}(x) dx = \delta_{\lambda,\tilde{\lambda}}$ for all $\lambda, \tilde{\lambda} \in \mathcal{J}$ with $|\lambda|, |\tilde{\lambda}| > 0$.

Tensorization of the univariate functions then allows for appropriate bases in higher dimensions as well as for a vast range of Bochner spaces arising in the formulation of parabolic PDEs, see e.g. [21]. Constructions for more complicated domains Ω are also available.

Then, we equivalently formulate (A.1) as the discrete, but infinite-dimensional equation

(A.4) Find
$$\mathbf{x} \in \ell_2(\widehat{\mathcal{J}})$$
: $\mathbf{A}\mathbf{x} = \mathbf{b}, \quad \mathbf{b} \in \ell_2(\check{\mathcal{J}}),$

where $\mathbf{A} := \langle \check{\mathbf{\Psi}}^{\mathcal{Y}}, \mathcal{A}[\widehat{\mathbf{\Psi}}^{\mathcal{X}}] \rangle$, $\mathbf{b} = [b(\check{\psi}^{\mathcal{Y}}_{\lambda})]_{\lambda \in \check{\mathcal{J}}}$ and \mathbf{x} are the coefficients of the (unique) expansion $x = \mathbf{x}^{\top} \widehat{\mathbf{\Psi}}^{\mathcal{X}}$.

A.2. Adaptive Methods and Nonlinear Approximation. In order to approximately solve the infinite-dimensional equation (A.4), AWGMs iteratively construct a sequence of *nested finite* index sets $(\hat{\Lambda}_k)_k \subset \hat{\mathcal{J}}$, $(\check{\Lambda}_k)_k \subset \check{\mathcal{J}}$, to which (A.4) is restricted. Considering (just for ease of presentation) a linear self-adjoint operator $\mathcal{A}[\cdot;\mu] : \mathcal{X} \to \mathcal{X}$ and $\hat{\Psi} = \hat{\Psi}^{\mathcal{X}} = \check{\Psi}^{\mathcal{Y}}$, in each iteration the finite-dimensional problem

(A.5) Find
$$\mathbf{x}_{\hat{\mathbf{\Lambda}}_k} \in \ell_2(\hat{\mathbf{\Lambda}}_k)$$
: $\hat{\mathbf{\Lambda}}_k \mathbf{A}_{\hat{\mathbf{\Lambda}}_k} \mathbf{x}_{\hat{\mathbf{\Lambda}}_k} = \mathbf{b}_{\hat{\mathbf{\Lambda}}_k}, \quad \mathbf{b}_{\hat{\mathbf{\Lambda}}_k} \in \ell_2(\hat{\mathbf{\Lambda}}_k),$

is solved, where for $\Lambda \subset \mathcal{J}$, $\mathbf{v}_{\Lambda} := \mathbf{v}|_{\Lambda}$ denotes the restriction of $\mathbf{v} \in \ell_2(\mathcal{J})$ to $\ell_2(\Lambda)$ and ${}_{\Lambda}A_{\Lambda} := (\mathbf{AE}_{\Lambda})|_{\Lambda}$ with trivial embedding $\mathbf{E} : \ell_2(\Lambda) \to \ell_2(\mathcal{J})$ the restriction of \mathbf{A} in both rows and columns.

The extension of Λ_k to Λ_{k+1} is then based on the residual $\mathbf{r}^k := \mathbf{b} - \mathbf{A}\mathbf{x}_{\Lambda_k}$ and its norm $\|\mathbf{r}^k\|_{\ell_2(\mathcal{J})}$ which forms an equivalent error estimator, since

(A.6)
$$\|\mathbf{A}\|^{-1} \|\mathbf{r}^{k}\|_{\ell_{2}(\hat{\mathcal{J}})} \leq \|\mathbf{x} - \mathbf{x}_{\hat{\mathbf{A}}_{k}}\|_{\ell_{2}(\hat{\mathcal{J}})} \leq \|\mathbf{A}^{-1}\| \|\mathbf{r}^{k}\|_{\ell_{2}(\hat{\mathcal{J}})}.$$

Note that \mathbf{r}^k is supported on the *infinite-dimensional* set $\hat{\mathcal{J}}$ even if \mathbf{x}_{Λ_k} is finitely supported. Hence we have to use appropriate approximation methods for the residual evaluation in order to arrive at an implementable AGWM, see §A.3 below.

The next index set is obtained by a so-called *bulk-chasing*: Choose $\Lambda_{k+1} \supset \Lambda_k$ as the smallest index set such that $\|\mathbf{r}_{\Lambda_{k+1}}^k\|_{\ell_2(\Lambda_{k+1})} \ge c\|\mathbf{r}^k\|_{\ell_2(\mathcal{J})}$ for some 0 < c < 1. This implies that the indices of the largest residual coefficients are added to Λ_k and the adaptive index set is steered into the direction of the largest error.

Under appropriate assumptions on the exactness and computational cost of the solution of (A.5), the approximation of \mathbf{r}^k and the implementation of the bulk chasing process, a *quasi-optimality* result is known. In order to formulate it, we introduce the nonlinear approximation class

(A.7)
$$\mathcal{A}^{s} := \left\{ \mathbf{v} \in \ell_{2}(\widehat{\mathcal{J}}) : \|\mathbf{v}\|_{\mathcal{A}^{s}} := \sup_{\varepsilon > 0} \varepsilon \left[\min\{\mathcal{N} \in \mathbb{N}_{0} : \|\mathbf{v} - \mathbf{v}_{\mathcal{N}}\|_{\ell_{2}(\widehat{\mathcal{J}})} \le \varepsilon \} \right]^{s} < \infty \right\}$$

with $\mathbf{v}_{\mathcal{N}}$ being the best \mathcal{N} -term approximation on \mathbf{v} , consisting of the \mathcal{N} largest coefficients in modulus of \mathbf{v} .

Theorem A.1 (cf. [10, 22]). There exist implementable routines and parameters such that the (approximate) computations of $\mathbf{x}_{\hat{\mathbf{\Lambda}}_k}$, \mathbf{r}^k and $\hat{\mathbf{\Lambda}}_{k+1}$ can be performed with controllable tolerances and computational cost: If the AWGM is terminated when $\|\mathbf{r}_{\hat{\mathbf{\Lambda}}_k}^k\|_{\ell_2(\hat{\mathbf{\Lambda}}_k)} \leq \varepsilon/\|\mathbf{A}^{-1}\|$, the output $\mathbf{x}_{\varepsilon} := \mathbf{x}_{\hat{\mathbf{\Lambda}}_k}$ satisfies $\|\mathbf{x} - \mathbf{x}_{\varepsilon}\|_{\ell_2(\hat{\mathcal{J}})} \leq \varepsilon$. If, moreover, $\mathbf{x} \in \mathcal{A}^s$ for some s > 0, it holds for $\mathcal{N}_k := \#\hat{\mathbf{\Lambda}}_k$ that

(A.8)
$$\|\mathbf{x} - \mathbf{x}_{\varepsilon}\|_{\ell_{2}(\hat{\mathcal{J}})} \leq C \|\mathbf{x}\|_{\mathcal{A}^{s}}^{1/s} \mathcal{N}_{k}^{-s}, \quad \# \operatorname{supp} \mathbf{x}_{\varepsilon} \leq C \varepsilon^{-1/s} \|\mathbf{x}\|_{\mathcal{A}^{s}}^{1/s}.$$

If s is small enough, the computation of \mathbf{x}_{ε} can be realized with a computational cost that is bounded by an absolute multiple of $\varepsilon^{-1/s} \|\mathbf{x}\|_{A^s}^{1/s}$, i.e., linear complexity. \Box

Theorem A.1 states that AWGMs are quasi-optimal in the sense that the optimal convergence rate for best \mathcal{N} -term approximations of \mathbf{x} can be realized up to some constant within linear computational complexity. These techniques can be extended to problems that are neither symmetric nor positive-definite by considering the normal equations $\mathbf{A}^{\mathsf{T}}\mathbf{A}\mathbf{x} = \mathbf{A}^{\mathsf{T}}\mathbf{b}$. This includes Petrov-Galerkin problems as they arise e.g. in space-time formulations of parabolic PDEs, even if the wavelet bases $\hat{\Psi}^{\mathcal{X}}, \, \check{\Psi}^{\mathcal{Y}}$ for \mathcal{X} and \mathcal{Y} differ not only in scaling but are even obtained from different sets of wavelets [5, 14].

A.3. Multitree-based Implementations. Several different implementations of quasi-optimal AWGMs have been proposed. The algorithms in [6, 7] use a thresholding step in order to retrieve the optimal computational complexity in Theorem A.1, which in the case of [7] is combined with an inexact Richardson iteration on the infinite-dimensional equation (A.4). In [10] a residual approximation method is employed that does not require thresholding and can thus be proven to be more efficient. However, like the afore-mentioned algorithms it relies on the application of a so-called **APPLY** routine in order to approximate the arising infinite-dimensional matrix-vector products $\mathbf{Av} \in \ell_2(\hat{\mathcal{J}})$. Such routines are based on wavelet compression schemes, require certain characteristics of the wavelet bases as well as compressibility results for the operator \mathcal{A} and are in general quantitatively demanding. For these reasons, we employ multitree-based matrix-vector product evaluations in the solution of (A.5) and the approximation of the residual \mathbf{r}^k , as proposed in

[15, 16]. That is, we restrict the index sets $\widehat{\Lambda}_k$ to multitrees in the sense of the following definition.

Definition A.2. (i) For a univariate uniformly local, piecewise polynomial wavelet basis $\Psi = \{\psi_{\lambda} : \lambda \in \mathcal{J}\}$, a set $\Lambda \subset \mathcal{J}$ is called a *tree* if for any $\lambda \in \Lambda$ with $|\lambda| > 0$ it holds that supp $\psi_{\lambda} \subset \bigcup_{\mu \in \Lambda; |\mu| = \lambda - 1} \operatorname{supp} \psi_{\mu}$. (ii) An index set $\Lambda \in \mathcal{J}$ belonging to a tensor product wavelet basis $\Psi = \{\psi_{\lambda} : \lambda \in \mathcal{J}\}$ is called a *multitree* if for all $i \in \{0, \ldots, n\}$ and all indices $\mu_j \in \mathcal{J}^{(j)}$ for $j \neq i$, the index set

(A.9)
$$\Lambda^{(i)} := \{\lambda_i \in \mathcal{J}^{(i)} : (\mu_0, \dots, \mu_{i-1}, \lambda_i, \mu_{i+1}, \dots, \mu_n) \in \mathbf{\Lambda}\} \subset \mathcal{J}^{(i)}$$

is either the empty set or a tree.

The restriction to such index sets preserves the quasi-optimality of the AWGM [15] in the constrained approximation class $\mathcal{A}^s_{\text{mtree}}$ defined w.r.t. $\|\mathbf{v}\|_{\mathcal{A}^s_{\text{mtree}}} := \sup_{\varepsilon > 0} \varepsilon \cdot \left[\min\{\mathcal{N} \in \mathbb{N}_0 : \|\mathbf{v} - \mathbf{v}_{\mathcal{N}}\|_{\ell_2(\widehat{\mathcal{J}})} \le \varepsilon \land \sup \mathbf{v}_{\mathcal{N}} \text{ is a multitree}\}\right]^s$ and allows a computationally very efficient evaluation of finite-dimensional matrix-vector products:

Theorem A.3 ([16, Theorem 3.1]). Let \mathcal{A} be a linear differential operator with polynomial coefficients and let $\hat{\Lambda} \subset \hat{\mathcal{J}}$, $\check{\Lambda} \in \check{\mathcal{J}}$ be multitrees. Then, for any $\mathbf{v}_{\hat{\Lambda}} \in \ell_2(\hat{\Lambda})$, the product ${}_{\check{\Lambda}}\mathbf{A}_{\hat{\Lambda}}\mathbf{v}_{\hat{\Lambda}}$ can be computed in $\mathcal{O}(\#\hat{\Lambda} + \#\check{\Lambda})$ operations.

Moreover, we obtain the following approximation result for the residual:

Theorem A.4 ([15]). Let $0 < \omega < 1$, let \mathcal{A} be a differential operator with polynomial coefficients and let $\mathbf{x} \in \mathcal{A}^s_{\text{mtree}}$ for some s > 0. Then, for all finite multitrees $\widehat{\mathbf{\Lambda}} \subset \widehat{\mathcal{J}}$ and all $\mathbf{w}_{\widehat{\mathbf{\Lambda}}} \in \ell_2(\widehat{\mathbf{\Lambda}})$, there exists a multitree $\check{\mathbf{\Xi}} = \check{\mathbf{\Xi}}(\widehat{\mathbf{\Lambda}}, \omega) \subset \check{\mathcal{J}}$ such that for $\mathbf{r} := \mathbf{b}_{\underline{\Xi}} - \underline{\mathbf{\Xi}} \mathbf{A}_{\widehat{\mathbf{\Lambda}}} \mathbf{w}_{\widehat{\mathbf{\Lambda}}}$ it holds that $\# \check{\mathbf{\Xi}} \leq C \# \widehat{\mathbf{\Lambda}} + \|\mathbf{r}\|^{-1/s}_{\ell_2(\check{\mathcal{J}})}$ and

(A.10)
$$\|(\mathbf{b} - \mathbf{A}\mathbf{w}_{\hat{\boldsymbol{\Lambda}}}) - \mathbf{r}\|_{\ell_{2}(\check{\boldsymbol{\mathcal{J}}})} \leq \omega \|\mathbf{r}\|_{\ell_{2}(\check{\boldsymbol{\mathcal{J}}})}.$$

Thus, the computational cost for the residual approximation is of the order $\mathcal{O}(\#\hat{\Lambda} + \|\mathbf{r}\|_{\ell_2(\check{\sigma})}^{-1/s})$ if the right hand side coefficients $\mathbf{b}_{\check{\Xi}}$ can be computed efficiently.

Explicit constructions of Ξ are discussed in [15] and [14], where the multitree-based AWGM is extended to the normal equations. In particular, such AWGM satisfies the conditions posed for the routine **SOLVE** in Section 3. We used AWGM for all adaptive computations (snapshots, Riesz representations, error estimates).

References

- [1] I. Babuška. Error-bounds for finite element method. Numer. Math., 16:322–333, 1970/1971.
- [2] M. Barrault, Y. Maday, N.C. Nguyen, and A.T. Patera. An empirical interpolation method: application to efficient reduced-basis discretization of partial differential equations. C. R. Acad. Sci. Paris, Ser. I, 339(9):667 – 672, 2004.
- [3] P. Binev, A. Cohen, W. Dahmen, R. DeVore, G. Petrova, and P. Wojtaszczyk. Convergence rates for greedy algorithms in reduced basis methods. SIAM Journal on Mathematical Analysis, 43(3):1457–1472, 2011.
- [4] T. Bui-Thanh, K. Willcox, and O. Ghattas. Model reduction for large-scale systems with high-dimensional parametric input space. SIAM J. Sci. Comput., 30(6):3270–3288, 2008.
- [5] N. G. Chegini and R. P. Stevenson. Adaptive wavelet schemes for parabolic problems: sparse matrices and numerical results. SIAM J. Numer. Anal., 49(1):182–212, 2011.
- [6] A. Cohen, W. Dahmen, and R. DeVore. Adaptive wavelet methods for elliptic operator equations: convergence rates. *Math. Comput.*, 70(233):27–75, 2001.

- [7] A. Cohen, W. Dahmen, and R. DeVore. Adaptive wavelet methods II beyond the elliptic case. Found. Comput. Math., 2:203–245, 2002.
- [8] R. Dautray and J.L. Lions. Mathematical analysis and numerical methods for science and technology. Evolution Problems I, volume 5. Springer-Verlag, Berlin, 1992.
- [9] T. Dijkema. Adaptive tensor product wavelet methods for solving PDEs. PhD thesis, Universiteit Utrecht, 2009.
- [10] T. Gantumur, H. Harbrecht, and R. P. Stevenson. An optimal adaptive wavelet method without coarsening of the iterands. *Math. Comput.*, 76(258):615–629, 2007.
- [11] M.A. Grepl and A.T. Patera. A posteriori error bounds for reduced-basis approximations of parameterized parabolic partial differential equations. ESAIM: Mathematical Modelling and Numerical Analysis, 39(1):157–181, 2005.
- [12] B. Haasdonk and M. Ohlberger. Reduced basis method for finite volume approximations of parameterized linear evolution equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 42:277–302, 2008.
- [13] S. Kestler. On the adaptive tensor product wavelet Galerkin method with applications in finance. PhD thesis, University of Ulm, 2013.
- [14] S. Kestler, K. Steih, and K. Urban. An efficient space-time wavelet Galerkin method for time-periodic parabolic partial differential equations. submitted, 2013.
- [15] S. Kestler and R. P. Stevenson. An efficient approximate residual evaluation in the adaptive tensor product wavelet method. J. Sci. Comp., 2013.
- [16] S. Kestler and R. P. Stevenson. Fast evaluation of system matrices w.r.t. multi-tree collections of tensor product refinable basis functions. accepted for publication in J. Comput. Appl. Math., 2013.
- [17] J. Nečas. Sur une méthode pour résoudre les équations aux dérivées partielles du type elliptique, voisine de la variationnelle. Ann. Scuola Norm. Sup. Pisa (3), 16:305–326, 1962.
- [18] R.H. Nochetto, K.G. Siebert, and A. Veeser. Theory of adaptive finite element methods: an introduction. In *Multiscale, nonlinear and adaptive approximation*, pages 409–542. Springer, Berlin, 2009.
- [19] 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. *Archives of Computational Methods in Engineering*, 15(3):229–275, 2008.
- [20] A. Rupp. High Dimensional Wavelet Methods for Structures Financial Products. PhD thesis, University of Ulm, (2013).
- [21] C. Schwab and R. P. Stevenson. Space-time adaptive wavelet methods for parabolic evolution problems. Math. Comput., 78(267):1293–1318, 2009.
- [22] R. P. Stevenson. Adaptive wavelet methods for solving operator equations: An overview. In R. DeVore and A. Kunoth, editors, *Multiscale, Nonlinear and Adaptive Approximation*, pages 543–598. Springer (Berlin), 2009.
- [23] K. Urban. Wavelet methods for elliptic partial differential equations. Oxford University Press, 2009.
- [24] K Urban, S. Volkwein, and O. Zeeb. Greedy sampling using nonlinear optimization. In A. Quarteroni and G. Rozza, editors, *Reduced Order Methods for modeling and computational reduction*, pages 137–157. Springer Switzerland, 2014.
- [25] M. Yano. A Reduced Basis Method with Exact-Solution Certificates for Symmetric Coercive Equations. Technical Report 64, MIT (USA), MechE, 2013. http://augustine.mit.edu/ methodology/methodology_technical_papers.html.

KRISTINA STEIH, INSTITUTE FOR NUMERICAL MATHEMATICS, UNIVERSITY OF ULM, HELMHOLTZ-STRASSE 20, D-89069 ULM, GERMANY

E-mail address: kristina.steih@uni-ulm.de

KARSTEN URBAN, INSTITUTE FOR NUMERICAL MATHEMATICS, UNIVERSITY OF ULM, HELM-HOLTZSTRASSE 20, D-89069 ULM, GERMANY

E-mail address: karsten.urban@uni-ulm.de