Electronic Transactions on Numerical Analysis.
Volume 32, pp. 145-161, 2008.

ISSN 1068-9613.

ETNA
Kent State University
etna@mcs.kent.edu

A REDUCED BASIS METHOD FOR EVOLUTION SCHEMES WITH PARAMETER-DEPENDENT EXPLICIT OPERATORS*

B. HAASDONK†, M. OHLBERGER‡, AND G. ROZZA§

Abstract. During the last decades, reduced basis (RB) methods have been developed to a wide methodology for model reduction of problems that are governed by parametrized partial differential equations (P^2DEs). In particular equations of elliptic and parabolic type for linear, low degree polynomial or monotonic nonlinearities have been treated successfully by RB methods using finite element schemes. Due to the characteristic offline-online decomposition, the reduced models often become suitable for a multi-query or real-time setting, where simulation results, such as field-variables or output estimates, can be approximated reliably and rapidly for varying parameters. In the current study, we address a certain class of time-dependent evolution schemes with explicit discretization operators that are arbitrarily parameter dependent. We extend the RB methodology to these cases by applying the *empirical interpolation* method to localized discretization operators. The main technical ingredients are: (i) generation of a *collateral reduced basis* modelling the effects of the discretization operator under parameter variations in the offline-phase and (ii) an online simulation scheme based on a numerical subgrid and localized evaluations of the evolution operator. We formulate an a-posteriori error estimator for quantification of the resulting reduced simulation error. Numerical experiments on a parametrized convection problem, discretized with a finite volume scheme, demonstrate the applicability of the model reduction technique. We obtain a parametrized reduced model, which enables parameter variation with fast simulation response. We quantify the computational gain with respect to the non-reduced model and investigate the error convergence.

Key words. model reduction, reduced basis methods, parameter dependent explicit operators, empirical interpolation, a-posteriori error estimates

AMS subject classifications. 76M12, 65M15, 35L90, 35K90, 76R99

1. Introduction.

General parametrized evolution problems for an unknown function $u(x,t;\boldsymbol{\mu}) : \Omega \times [0,T] \to \mathbb{R}$ depending on a parameter $\boldsymbol{\mu} \in \mathcal{P} \subset \mathbb{R}^p$ can frequently be found in the form of a parametrized partial differential equation (P^2DE) for u,

$$\partial_t u(\cdot,t;\boldsymbol{\mu}) + \mathcal{L}(\boldsymbol{\mu},t)[u(\cdot,t;\boldsymbol{\mu})] = 0,$$

with corresponding initial and boundary conditions, where $[\cdot]$ denotes the evaluation of the spatial differential operator. The parameter domain, where the parameter vector $\boldsymbol{\mu}$ stems from, is denoted by $\mathcal{P} \subset \mathbb{R}^p$. The initial data, denoted by $u_0(x;\boldsymbol{\mu})$, and the solution commonly have some spatial regularity $u_0(\cdot;\boldsymbol{\mu}), u(\cdot,t;\boldsymbol{\mu}) \in \mathcal{W}$. Numerical treatment of such evolution problems is frequently based on a time discretization at a finite number of time instances $0 = t^0 < \ldots < t^K = T$ by finite differences or higher order Runge-Kutta type time integration. For the space discretization a finite but frequently high dimensional space $\mathcal{W}_H$ for approximating the solution at the discrete times is available, i.e., $u(\cdot,t^k;\boldsymbol{\mu}) \approx u_H(\cdot;\boldsymbol{\mu},t^k) \in \mathcal{W}_H$, where $H := \dim(\mathcal{W}_H)$. Typically, this is a finite element (FE), finite volume (FV) or discontinuous Galerkin (DG) space.

The motivation for reduced basis (RB) methods is founded on the need to solve a given P^2DE repeatedly in a multi-query setting such as parameter variation for design, optimization,

*Received November 29, 2007. Accepted for publication July 2, 2008. Published online on March 21, 2009. Recommended by A. Meyer.

†Abteilung für Angewandte Mathematik, Albert-Ludwigs-Universität Freiburg, Hermann-Herder-Str. 10, 79104 Freiburg, Germany. Also at Institut für Numerische und Angewandte Mathematik, Westfälische Wilhelms-Universität Münster, Einsteinstr. 62, 48149 Münster, Germany (haasdonk@math.uni-muenster.de).

‡Institut für Numerische und Angewandte Mathematik, Westfälische Wilhelms-Universität Münster, Einsteinstr. 62, 48149 Münster, Germany (mario.ohlberger@math.uni-muenster.de).

§Department of Mechanical Engineering, Massachusetts Institute of Technology, 77 Massachusetts Avenue, Cambridge, MA 02139 (rozza@mit.edu).

control, inverse problems or statistical analysis. In the following we abbreviate $u_H^k(\boldsymbol{\mu}) := u_H(\cdot, t^k; \boldsymbol{\mu})$.

Numerical evolution schemes of first order mostly consist of implicit and explicit contributions, which compute the sequence $u_H^k(\boldsymbol{\mu}), k = 0, \dots, K$, by starting with a suitable projection $P : \mathcal{W} \to \mathcal{W}_H$ of the given initial data $u_H^0(\boldsymbol{\mu}) := P[u_0(\cdot; \boldsymbol{\mu})]$ and successively solving the following equation for $u_H^{k+1}(\boldsymbol{\mu})$, $k = 1, \dots, K-1$:

$$\frac{1}{\Delta t}\left(u_H^{k+1}(\boldsymbol{\mu}) - u_H^k(\boldsymbol{\mu})\right) + \mathcal{L}_I(\boldsymbol{\mu}, t^k)[u_H^{k+1}(\boldsymbol{\mu})] + \mathcal{L}_E(\boldsymbol{\mu}, t^k)[u_H^k(\boldsymbol{\mu})] = 0. \tag{1.1}$$

Here $\mathcal{L}_I(\boldsymbol{\mu}, t^k),\ \mathcal{L}_E(\boldsymbol{\mu}, t^k) : \mathcal{W}_H \to \mathcal{W}_H$ denote the implicit and explicit discretization contributions of the analytical spatial differential operator $\mathcal{L}(\boldsymbol{\mu}, t^k)$.

A general description of the Reduced Basis methodology for stationary cases can be found in [10, 13]. Time dependent problems are for instance treated in [2, 11]. The general goal in case of time-dependence is to find a sequence of functions $u_N^k(\boldsymbol{\mu}), k = 0, \dots, K$, in a *reduced basis space* $\mathcal{W}_N \subset \mathcal{W}_H$ of low dimension $N \ll H$, which approximates the detailed solution sequence, i.e., $u_N^k(\boldsymbol{\mu}) \approx u_H^k(\boldsymbol{\mu})$. In particular, the complexity of the computation scheme for determining these *reduced basis* solutions should be independent of H. In addition to this general goal further questions in RB methods deal with general outputs $s(u_H)$ derived from the field variable and their RB estimation. For many problems, the provision of effective a-posteriori error estimators is a distinctive feature of RB methods.

Special instances of evolution schemes of type (1.1) have been treated in the literature with RB methods: The case of a pure implicit FE space discretization, i.e., $\mathcal{L}_E \equiv 0$, and affine $\mathcal{L}_I$ was treated in [4]. The extension to the case of nontrivial explicit operators, e.g., covering FV schemes, while the operators still are assumed to be affine in u, was formulated in [6]. The parabolic case for a monotonic pointwise nonlinearity was treated in [2, 3].

In the current study we devise an RB formulation for the pure explicit case. That means we confine ourselves to the case $\mathcal{L}_I(\boldsymbol{\mu}, t^k) \equiv 0$ and $\mathcal{L}_E(\boldsymbol{\mu}, t^k)$ being a general parameter dependent operator with a certain localized representation. This localized structure allows us to apply the *empirical interpolation* technique [1] to approximate the operator evaluations.

In the next section we specify the class of explicit discretization operators that can be approximated with our approach and we present the reduced simulation scheme. The reduced simulation scheme requires a decomposition of the computation in an offline and online-part. We describe details on this decomposition in Section 3. As an analytical result, we present an a-posteriori error estimator in Section 4. Experiments in Section 5 on a simple convection model indicate the applicability of the method. In particular, we investigate the computational gain and the error convergence. We conclude our study in Section 6.

2. RB approximation for explicit evolution schemes. In this section we will formulate the RB approximation for the class of evolution schemes that we are interested in. For this we will first give some general definitions, such that we can specify our assumptions on the discretization space $\mathcal{W}_H$ and the discretization operators.

DEFINITION 2.1 (Local Basis of $\mathcal{W}_H$). *Let $\Psi := \{\psi_i | i = 1, \dots, H\}$ be the basis of $\mathcal{W}_H$ on which the evolution scheme and space-discretization is based. For a function $\phi \in \mathcal{W}_H$, we denote $(\phi)_i$ to be the coefficient or* degree of freedom *(DOF) corresponding to ψ_i in the basis expansion $\phi = \sum_{i=1}^H (\phi)_i \psi_i$. The set of basis function indices that support the value of functions $\phi \in \mathcal{W}_H$ at a given point $x \in \Omega$ is denoted by $I(x) = \{i | \psi_i \in \Psi \text{ and } \psi_i(x) \neq 0\}$. We call Ψ a* local basis *if the size of these index sets is bounded by a constant independent of H, i.e., there exists a J, such that $\operatorname{card}(I(x)) \leq J$ for almost all $x \in \Omega$. For any set of DOF-indices $S \subset \{1, \dots, H\}$, we further define the projection $\pi_S : \mathcal{W}_H \to \operatorname{span}\{\psi_i \in \Psi | i \in S\}$ on the corresponding subspace via $\pi_S(\psi_i) = \psi_i$ for all $i \in S$ and $\pi_S(\psi_i) = 0$, otherwise.*

ETNA
Kent State University
etna@mcs.kent.edu

Finally, let $X_H \subset \Omega$ *be a set of* H *points, such that the restricted functions* $\psi_i|_{X_H}$ *allow pointwise evaluations and are linearly independent.*

Trivially, J is bounded by H. But our approximation scheme will depend on the fact that the basis is a local basis in the sense that this number J is much smaller and independent of H. This is typical for FE or FV basis functions, which have support only in few grid elements, and the number J only depends on the shape of the grid elements and not on the overall number of cells. For convenience, we exemplify the notation for a FV space in 2 dimensions, as this also will be used in the experiments.

EXAMPLE 2.2 (Local Basis of Finite Volume Space). *Assume a non-degenerate triangulation* $\mathcal{T} = \{T_i\}$ *of a bounded polygonal domain* $\Omega \subset \mathbb{R}^2$ *with* H *disjoint triangles* $T_i, i = 1, \ldots, H$. *After choosing an elementwise constant Ansatz space, a corresponding basis is simply given by the indicator functions* $\psi_i := \chi_{T_i}$ *of the triangles. We easily verify that this is a local basis in the sense of Definition 2.1: For* $x \in T_i$ *only the single basis function* ψ_i *has support in* x*, hence* $I(x) = \{i\}$ *with upper bound on the cardinality* $J := 1$*. In particular, this value is independent of* H*. As the point set* $X_H \subset \Omega$ *we simply can use the centroids of the triangles,* $X_H := \{c(T_i)|i = 1, \ldots, H\}$*, as the values of piecewise constant functions are well defined in the triangle centers and the restrictions* $\psi_i|_{X_H}$ *are linearly independent.*

The main requirement for efficient approximation of evolution schemes is a small stencil of the discretization operators. Intuitively, this means that the value of $\mathcal{L}_E(\boldsymbol{\mu}, t^k)[u]$ in a certain point in space only depends on a small ($\boldsymbol{\mu}$, t^k and H-independent) number of at most J_E point-evaluations of u, or more generally, only J_E DOFs of u. For this, we will assume the following general structure of the explicit discretization operator, which will be relevant for efficient approximation.

DEFINITION 2.3 (Localized Discretization Operator). *A discretization operator* $\mathcal{L}_E : \mathcal{W}_H \to \mathcal{W}_H$ *can be expressed as*

$$\mathcal{L}_E[u] := \sum_{i=1}^{H} l_i(u)\psi_i, \tag{2.1}$$

with suitable functionals $l_i : \mathcal{W}_H \to \mathbb{R}$*, which represent the coefficients of the operator evaluation. Each of these functionals* l_i *has a set of DOFs* $S_i \subset \{1, \ldots, H\}$ *on which it depends and all other DOFs do not influence the result, i.e.,*

$$l_i(u) = l_i(\pi_{S_i}[u]) \quad \textit{for all} \quad u \in \mathcal{W}_H. \tag{2.2}$$

We call the operator $\mathcal{L}_E$ *a* localized operator *if there exists a constant* J_E *independent of* H *with* $\mathrm{card}(S_i) \leq J_E$ *for all* i *and the computation of a single* $l_i(\pi_{S_i}[u])$ *has complexity polynomial in* J_E*, i.e.,* $\mathcal{O}(J_E^\alpha)$ *for a small integer* α*.*

Again, J_E is trivially bounded by H. In the following, however, the computational gain will depend on a small value of $J_E \ll H$. For example, first order FV operators are localized operators in this sense. This is illustrated in the following example.

EXAMPLE 2.4 (Finite Volume Discretization Operator as Localized Operator). *Assume a conform triangulation and a piecewise constant function space as given in Example 2.2. Let* $u_H^k = \sum_{i=1}^H u_i^k \psi_i \in \mathcal{W}_H$ *denote a piecewise constant function at time* k *with elementwise values* $u_i^k \in \mathbb{R}$*. A first order explicit finite volume time evolution for computing* $u_H^{k+1} = \sum_{i=1}^H u_i^{k+1}\psi_i$ *is defined by*

$$u_i^{k+1} = u_i^k - \frac{\Delta t}{|T_i|} \sum_{j \in \mathcal{N}(i)} g_{ij}(u_i^k, u_j^k), \tag{2.3}$$

where $|T_i|$ denotes the area of triangle T_i, $\mathcal{N}(i)$ denotes the set of triangle-indices that share an edge with T_i, and $g_{ij}(u,v) : \mathbb{R}^2 \to \mathbb{R}$ denotes a suitable choice of numerical flux functions [7]. For simplification of the presentation, we omit the boundary treatment here. Comparing (2.3) with (1.1), we can identify $\mathcal{L}_I = 0$ and $\mathcal{L}_E[u_H^k] = \sum_{i=1}^H l_i(u_H^k)\psi_i$ by the coefficient functionals

$$l_i(u_H^k) := \frac{1}{|T_i|} \sum_{j \in \mathcal{N}(i)} g_{ij}(u_i^k, u_j^k).$$

In particular, the value of $l_i(u_H^k)$ only depends on the element values on T_i and its neighbours. Consequently, the sets of DOF-dependency are simply given by $S_i = \mathcal{N}(i) \cup \{i\}$. Obviously every triangle has a maximum of 3 neighbours, hence $\mathrm{card}(S_i) \leq J_E := 4$ independent of H and the computation of the functionals $l_i(u_H^k)$ has complexity linear in J_E.

Another example of localized operators are FE operators using basis functions with small support, e.g., nodal bases, where the number J_E is the maximum number of basis functions, which have common support on some mesh element.

The *empirical interpolation method* [1] was proposed for approximation of non-affinely parameter dependent or nonlinear analytical functions which allows a fast online-interpolation scheme. Geometry variation was treated in RB literature, e.g., [8], and in particular by this interpolation scheme [12, 14]. We will adopt this procedure to approximate discretization operator evaluations.

DEFINITION 2.5 (Empirical Interpolation of Localized Operator Evaluations). *For a localized discretization operator $\mathcal{L}_E(\boldsymbol{\mu}, t^k)$ we assume to have given a ($\boldsymbol{\mu}$ and t^k-independent) collateral basis space $\mathcal{W}_M \subset \mathcal{W}_H$ of dimension M, spanned by* snapshots *of the operator evaluation $\mathcal{W}_M := \mathrm{span}\{\mathcal{L}_E(\boldsymbol{\mu}_i, t^{k_i})[u_H^{k_i}(\boldsymbol{\mu}_i)] | i = 1, \dots, M\}$ for suitably chosen $\boldsymbol{\mu}_i$ and k_i. We further assume the availability of a set of interpolation points $T_M := \{x_1, \dots, x_M\} \subset X_H$ in Ω and corresponding nodal basis $\boldsymbol{\xi}_M := \{\xi_1, \dots, \xi_M\} \subset \mathcal{W}_M$ satisfying $\xi_j(x_i) = \delta_{ij}$. We denote the corresponding interpolation operator as $\mathcal{I}_M : \mathcal{W}_H \to \mathcal{W}_M$, which is consequently given by $\mathcal{I}_M[v] = \sum_{m=1}^M v(x_m)\xi_m$ and satisfies $\mathcal{I}_M[v](x_m) = v(x_m)$ for all $m = 1, \dots, M$ and $v \in \mathcal{W}_H$.*

The DOF-index set that supports a numerical function in any of these points is given by $I_M := \bigcup_{x \in T_M} I(x) \subset \{1, \dots, H\}$, where $I(x)$ is given in Definition 2.1. The larger set of DOF-indices which are required for the computation of these target DOFs by the coefficient functionals is obtained as $S_M := \bigcup_{i \in I_M} S_i$.

For any given $\boldsymbol{\mu} \in \mathcal{P}, k \in \{0, \dots, K-1\}$ and $u \in \mathcal{W}_H$, we can determine the desired interpolation values in the interpolation points

$$y_m(u, \boldsymbol{\mu}, t^k) := \mathcal{L}_E(\boldsymbol{\mu}, t^k)[u](x_m) \quad \text{for} \quad m = 1, \dots, M, \tag{2.4}$$

and obtain the empirical interpolation of the operator evaluation as

$$\mathcal{I}_M[\mathcal{L}_E(\boldsymbol{\mu}, t^k)[u]] = \sum_{m=1}^M y_m(u, \boldsymbol{\mu}, t^k)\xi_m(x). \tag{2.5}$$

The motivation for the notion *empirical interpolation* stems from the fact that the collateral reduced basis space $\mathcal{W}_M$ is constructed from simulation results, i.e. "empirical" data. The construction of the interpolation basis $\boldsymbol{\xi}_M$ and interpolation points T_M will be adressed in the subsequent section. The key observation for the use of the empirical interpolation in RB methods is that it results in an effective separation of space-independent coefficients

ETNA
Kent State University
etna@mcs.kent.edu

$y_i(u, \boldsymbol{\mu}, t^k)$ and parameter-independent functions $\xi_i(x)$. This enables an efficient offline-online decomposition.

The core of a (Lagrangian) reduced basis approximation of the evolution equation (1.1) with $\mathcal{L}_I \equiv 0$ is the availability of a reduced basis space $\mathcal{W}_N \subset \mathcal{W}_H$ of low dimension $N := \dim \mathcal{W}_N$ constructed from snapshots of the unknown field variable $u_H^{k_i}(\boldsymbol{\mu}_i)$ for suitable parameters $\boldsymbol{\mu}_i$ and time-indices k_i. These parameters may very well be different from the ones used for constructing the collateral reduced basis space $\mathcal{W}_M$. For computational reasons, it is beneficial to work with an orthonormal basis $\Phi_N := \{\varphi_1, \ldots, \varphi_N\}$ of $\mathcal{W}_N$. Here, proper orthogonal decomposition methods could be used, which result in an orthonormal basis by construction. A Galerkin projection of the explicit evolution scheme onto this subspace leads to the following weak formulation of the problem: start with a suitable projection of the initial data by determining $u_N^0(\boldsymbol{\mu}) \in \mathcal{W}_N$, such that

$$(u_N^0(\boldsymbol{\mu}), v) = (P[u_0(\boldsymbol{\mu})], v) \quad \forall v \in \mathcal{W}_N,$$

and then subsequentially find $u_N^{k+1}(\boldsymbol{\mu}) \in \mathcal{W}_N$ for all $k = 0, \ldots, K-1$, such that

$$\big(u_N^{k+1}(\boldsymbol{\mu}), v\big) + \big((\Delta t \mathcal{L}_E(\boldsymbol{\mu}, t^k) - Id)[u_N^k(\boldsymbol{\mu})], v\big) = 0 \quad \forall v \in \mathcal{W}_N. \tag{2.6}$$

Here Id denotes the identity operator.

For an effective offline-online decomposition in the next section, we will additionally require a so called *affine parameter-dependence* of the initial data, i.e.,

$$u_0(\boldsymbol{\mu}) = \sum_{q=1}^{Q_{u_0}} \sigma_{u_0}^q(\boldsymbol{\mu}) u_0^q(x), \tag{2.7}$$

with a small number Q_{u_0} of parameter-independent functions $u_0^q(x)$ and space-independent coefficients $\sigma_{u_0}^q(\boldsymbol{\mu})$. If this decomposition is not available in a given model-problem, an additional empirical interpolation of the initial data can provide an arbitrarily accurate approximation. If we replace the evaluations $\mathcal{L}_E(\boldsymbol{\mu}, t^k)[u_N^k]$ by the empirical interpolations $\mathcal{I}_M[\mathcal{L}_E(\boldsymbol{\mu}, t^k)[u_N^k]]$, we can formulate the RB approximation of the explicit evolution scheme as follows:

DEFINITION 2.6 (Reduced Basis Approximation with Empirical Interpolation of $\mathcal{L}_E$). *We assume that we have given an explicit evolution scheme, where $\mathcal{L}_E(\boldsymbol{\mu}, t^k)$ is assumed to be an arbitrary explicit discretization operator. We assume that an appropriate empirical interpolation scheme is defined by means of interpolation basis $\boldsymbol{\xi}_M$ and interpolation points $T_M \subset \Omega$, and a reduced basis Φ_N is available. We then define the following scheme for sequentially computing $u_N^k(\boldsymbol{\mu}) := \sum_n a_n^k(\boldsymbol{\mu})\varphi_n$ by specifying its coefficient vectors $\mathbf{a}^k = (a_1^k, \ldots, a_N^k)^T \in \mathbb{R}^N$ for $k = 0, \ldots, K$:*

$$\mathbf{a}^0 := \left((P[u_0(\boldsymbol{\mu})], \varphi_1), \ldots, (P[u_0(\boldsymbol{\mu})], \varphi_N)\right)^T, \tag{2.8}$$

$$\mathbf{a}^{k+1} = \mathbf{a}^k - \Delta t \mathbf{C}_E \mathbf{l}_E(\boldsymbol{\mu}, t^k)[\mathbf{a}^k]. \tag{2.9}$$

Here, the corresponding vectors and matrices are defined as

$$(\mathbf{C}_E)_{nm} := (\xi_m, \varphi_n), \tag{2.10}$$

$$\left(\mathbf{l}_E(\boldsymbol{\mu}, t^k)[\mathbf{a}^k]\right)_m := \mathcal{I}_M[\mathcal{L}_E(\boldsymbol{\mu}, t^k)[u_N^k]](x_m), \tag{2.11}$$

for $n = 1, \ldots, N$ and $m = 1, \ldots, M$. The resulting sequence of functions $\{u_N^k(\boldsymbol{\mu})\}_{k=0}^K$ finally defines the reduced basis approximation *$u_N(\mathbf{x}, t; \boldsymbol{\mu})$ to coincide with $u_N^k(\mathbf{x}; \boldsymbol{\mu})$ in the time-slab $[t^k, t^{k+1})$.*

Due to the well-definedness of the empirical interpolation for a given vector $\mathbf{a}^k$ all quantities are uniquely defined.

3. Offline-online decomposition. A fundamental ingredient in reduced basis approximation of P^2DEs is the effective decomposition of the computations in an offline- and online-phase. The offline phase prepares parameter-independent quantities, the computation of which is (typically heavily) depending on H. The online phase assembles the final parameter-dependent matrices and vectors for the RB simulation, which is ideally independent of the complexity H.

3.1. Offline-phase. Certain quantities are computed in the offline-phase as they are independent of the parameter $\boldsymbol{\mu}$, which is only available in the online-stage. We discriminate between two steps in the offline-phase.

3.1.1. Offline-phase step 1. The first step derives possibly H-dependent quantities, which therefore may not be used in the online-simulation as such. This step is largely based on running detailed simulations for different parameters, and hereby contributes the dominating part to the computation time.

The empirical interpolation of the operator evaluation is the main new component in the RB scheme, though it largely follows the standard formulation of the *empirical interpolation* of functions [1]. To start, a set of snapshots of the operator evaluation is generated by

$$L_{train} = \{\mathcal{L}_E(\boldsymbol{\mu}, t^k)[u_H^k(\boldsymbol{\mu})] | k = 0, \ldots, K, \boldsymbol{\mu} \in M_{train}\} \subset \mathcal{W}_H, \tag{3.1}$$

for some finite training set $M_{train} \subset \mathcal{P}$. Thus, for each $\boldsymbol{\mu} \in M_{train}$ the whole trajectory $\{u_H^k(\boldsymbol{\mu})\}_{k=0}^K$ is contained in L_{train}. This dense sampling in time turned out to be necessary for good empirical interpolation of these trajectories. Now, for all $m = 1, \ldots, M$ (or an earlier stop at point 4, if a certain approximation accuracy on L_{train} is obtained), we consecutively determine functions $q_m \in \mathcal{W}_H$ and interpolation points $x_m \in X_H$ by the following, starting with $m = 1$:

1. Define $\mathcal{W}_{m-1} := \mathrm{span}\{q_i | i = 1, \ldots, m-1\}$ (with $\mathcal{W}_{m-1} := \{0\}$ if $m = 1$).
2. For all $v \in L_{train}$ determine the best approximation v^* in $\mathcal{W}_{m-1}$ by
$$v^* := \arg \min_{w \in \mathcal{W}_{m-1}} \|v - w\|_{L^2(\Omega)}^2 . \tag{3.2}$$
3. Determine the snapshot in L_{train} that has the worst error
$$v_m := \arg \max_{v \in L_{train}} \|v - v^*\|_{L^2(\Omega)} . \tag{3.3}$$
4. If the error for v_m is less than a prescribed ε_{tol} then stop the loop with $M := m-1$.
5. Otherwise, if $m > 1$ then solve the following equation system to obtain interpolation coefficients $\boldsymbol{\sigma}^{m-1} := (\sigma_j^{m-1})_{j=1}^{m-1} \in \mathbb{R}^{m-1}$,
$$\sum_{j=1}^{m-1} \sigma_j^{m-1} q_j(x_i) = v_m(x_i) \quad \text{for} \quad i = 1, \ldots, m-1. \tag{3.4}$$
6. Compute the residual function, i.e., error between v_m and its current interpolant
$$r_m := v_m - \sum_{j=1}^{m-1} \sigma_j^{m-1} q_j .$$
Note, in particular, that in case of $m = 1$ the sum is empty, so $r_m = v_m$.

7. Search the maximizer of r_m as new interpolation point and normalize to obtain the new interpolation function

$$x_m := \arg \sup_{x \in X_H} |r_m(x)|, \qquad q_m := r_m / r_m(x_m). \tag{3.5}$$

8. Increase $m := m + 1$ and if $m \leq M$ then repeat starting with step 1.

Note, that apart from different and slightly simplified notation the above algorithm is mainly the so called *surrogate* method [3] of empirical interpolation, as we use the L^2-error for choosing the worst approximation in (3.2) and (3.3). This is known to be computationally more efficient than L^∞-norm approximation, which requires a solution of a linear program for each training parameter vector in each extension step.

Differences to the formulation in [3] lie in the choice of the initial function, which is not random in our case, and in the restriction of the search space for the interpolation points $x_m \in X_H$. These are considered minor natural modifications for the case of dealing with discrete functions. It might happen, that the minimization/maximization operations have non-unique optima. In this case, refined selection criteria can be defined based on enumerations of the finite search spaces. In case of multiple maxima of (3.3), choose an enumeration of the set L and take the v_m from the set of worst approximated functions that has smallest index in the enumeration. In case of multiple maxima in (3.5), we again obtain a unique point involving an enumeration of the set. For example, in case of nodal basis functions, the maximization can be restricted to the set of these nodes. The set $\mathbf{Q}_M := \{q_1, \ldots, q_M\}$ of functions is a non-nodal basis for the interpolating space $\mathcal{W}_M$. We additionally introduced the nodal basis $\boldsymbol{\xi}_M$ in Definition 2.5. Formally both are equivalent, as they both span $\mathcal{W}_M$. Computationally the basis $\mathbf{Q}_M$ is used for all interpolation steps, whereas for ease of intuition, the subsequent argumentation mainly uses the nodal basis $\boldsymbol{\xi}_M$.

The remaining crucial quantity is the construction of a reduced basis Φ_N spanning $\mathcal{W}_N$. In RB approaches such schemes frequently are based on a given training set of parameters $M_{train} \subset \mathcal{P}$ and an incremental basis extension procedure involving a *greedy search* [10]. This means, given a current small reduced basis, reduced simulations are run for all parameters $\boldsymbol{\mu} \in M_{train}$, the parameter $\boldsymbol{\mu}^*$ with the worst error $\|u_H(\boldsymbol{\mu}^*) - u_N(\boldsymbol{\mu}^*)\|_{L^\infty(L^2)}$ (or estimate thereof) is determined, a new basis vector is constructed from the detailed simulation $u_H(\boldsymbol{\mu}^*)$, and the current reduced basis is extented by this. In the present study we choose the approach as described in detail in [6]. Instead of an a-posteriori error estimator, which was used there for estimating the error, we compute the true error as we have the detailed simulations $u_H(\boldsymbol{\mu})$ for all $\boldsymbol{\mu} \in M_{train}$ available. More sophisticated procedures such as adaptive training set extension can also be applied [5]. Note that such incremental reduced basis construction requires reduced simulations for assessing the quality of the current basis. So repeated evaluation of all subsequent online-steps in Section 3.2 and returning to this offline-step 1 is required until the final basis is obtained.

3.1.2. Offline-phase step 2. The above quantities are partially dependent on H, so the second step of the offline-phase provides the final quantities that are used in the subsequent online-simulation. Their computation may very well still be H-dependent, but the quantities themselves are independent of H and parameter independent.

- We compute component-vectors $\{\mathbf{a}_0^q\}_{q=1}^{Q_{u_0}}$ for the initial data by

$$\mathbf{a}_0^q := \left((P[u_0^q(\boldsymbol{\mu})], \varphi_1), \ldots, (P[u_0^q(\boldsymbol{\mu})], \varphi_N)\right)^T \qquad \text{for} \qquad q = 1, \ldots, Q_{u_0}.$$

- We compute the cross-gram-matrix $\mathbf{C}_E$ between the reduced basis Φ_N and the nodal basis $\boldsymbol{\xi}_M$ of the collateral space by (2.10).

- We define an arbitrary enumeration $\iota : \{1, \ldots, \mathrm{card}(I_M)\} \to I_M$ of the set I_M from Definition 2.5 and compute

$$\mathbf{J} \in \mathbb{R}^{M \times \mathrm{card}(I_M)} \quad \text{with} \quad (\mathbf{J})_{mj} = \psi_{\iota(j)}(x_m). \tag{3.6}$$

- The projections $\pi_{S_M}[\varphi_n]$ for all reduced basis vectors $\varphi_n \in \Phi_N$ are computed and stored in such a way that linear combinations can be computed efficiently with complexity $\mathcal{O}(\mathrm{card}(S_M)N)$, i.e., in particular H-independently. This can be realized by storing the Ψ-basis expansion coefficients $(\varphi_n)_i$ for $i \in S_M$ in a matrix along with a corresponding enumeration of the set S_M.
- Depending on the implementation of the localized operator, further numerical quantities may be required for the online stage. For example, in our implementation a numerical subgrid is extracted from the the detailed grid that contains the elements supporting the basis functions ψ_i for $i \in S_M$.

It can easily be verified, that the memory complexity of these quantities is independent of H, which is the basic requirement for an H-independent online-phase.

3.2. Online-phase. In the online-phase, the parameter $\boldsymbol{\mu} \in \mathcal{P}$ is specified and the offline-quantities are combined by H-independent operations to realize the RB approximation of Definition 2.6. The start of the simulation is quite obvious: The parameter dependent projection (2.8) is replaced by a linear combination of offline-quantities while making use of the affine parameter-dependence (2.7) of $u_0(\boldsymbol{\mu})$:

$$\mathbf{a}_0 = \sum_{q=1}^{Q_{u_0}} \sigma_{u_0}^q(\boldsymbol{\mu}) \mathbf{a}_0^q.$$

This is an overall operation of complexity $\mathcal{O}(Q_{u_0}N)$, independent of H. The main ingredient in the online-phase is the online-computation of the empirical interpolation in case of localized operators. This is the main new component of the present scheme.

PROPOSITION 3.1 (Online Empirical Interpolation). *We assume to have an explicit evolution scheme and corresponding RB approximation according to Definition 2.6. If the explicit operator $\mathcal{L}_E(\boldsymbol{\mu}, t^k)$ is a localized operator, then the computation of the empirical interpolation (2.11) for a coefficient vector $\mathbf{a}^k$ of a function $u_N^k = \sum_n a_n^k \varphi_n \in \mathcal{W}_N$ can be performed by the following steps:*

(i) Determine the partial reconstruction $v := \pi_{S_M}[u_N^k]$ of the RB solution by

$$v = \sum_{n=1}^{N} a_n^k \pi_{S_M}[\varphi_n]. \tag{3.7}$$

(ii) Let $l_i(\boldsymbol{\mu}, t^k)$ denote the parameter-dependent coefficient functionals of the localized representation (2.1) of $\mathcal{L}_E(\boldsymbol{\mu}, t^k)$ and compute these for $i \in I_M$ by

$$\mathbf{l}(\boldsymbol{\mu}, t^k) := (l_{\iota(j)}(\boldsymbol{\mu}, t^k)[v])_{j=1}^{\mathrm{card}(I_M)}. \tag{3.8}$$

(iii) Perform the interpolation by

$$\mathbf{l}_E(\boldsymbol{\mu}, t^k) = \mathbf{J}\mathbf{l}(\boldsymbol{\mu}, t^k), \tag{3.9}$$

with $\mathbf{J}$ given as in (3.6).

ETNA
Kent State University
etna@mcs.kent.edu

In particular, the complexity of these computations is independent of H, only polynomial in N, M, J and J_E.

Proof. We verify that the above computational scheme indeed results in the interpolation property (2.11). For this we first recall from (2.4) and (2.5) that for all $x \in \Omega$,

$$\mathcal{I}_M[\mathcal{L}_E(\boldsymbol{\mu}, t^k)[u_N^k]](x) = \sum_{m=1}^{M} \left(\mathcal{L}_E(\boldsymbol{\mu}, t^k)[u_N^k]\right)(x_m)\xi_m(x).$$

As ξ_m are nodal basis functions, an evaluation in an interpolation point $x_m \in T_M$ is obtained (abbreviating $l_i = l_i(\boldsymbol{\mu}, t^k)$) by

$$\mathcal{I}_M[\mathcal{L}_E(\boldsymbol{\mu}, t^k)[u_N^k]](x_m) = \left(\mathcal{L}_E(\boldsymbol{\mu}, t^k)[u_N^k]\right)(x_m) = \sum_{i=1}^{H} l_i(u_N^k)\psi_i(x_m).$$

As $\mathcal{L}_E$ is a localized discretization operator and $\psi_i(x_m) = 0$ for $i \notin I_M$, we obtain

$$\mathcal{I}_M[\mathcal{L}_E(\boldsymbol{\mu}, t^k)[u_N^k]](x_m) = \sum_{i \in I_M} l_i(u_N^k)\psi_i(x_m). \tag{3.10}$$

Definition 2.1 implies $\pi_{S_i} \circ \pi_{S_M} = \pi_{S_i}$ for $i \in I_M$. Then, using (2.2) yields

$$l_i(u_N^k) = l_i(\pi_{S_i}[u_N^k]) = l_i(\pi_{S_i} \circ \pi_{S_M}[u_N^k]) = l_i(\pi_{S_M}[u_N^k]) = l_i(v),$$

as v is defined by (3.7). Inserting this in (3.10), rewriting the summation, and using (3.8) and (3.9) yields

$$\mathcal{I}_M[\mathcal{L}_E(\boldsymbol{\mu}, t^k)[u_N^k]](x_m) = \sum_{j=1}^{\mathrm{card}(I_M)} l_{\iota(j)}(v)\psi_{\iota(j)}(x_m) = \left(\mathbf{J}\mathbf{l}(\boldsymbol{\mu}, t^k)\right)_m = \left(\mathbf{l}_E(\boldsymbol{\mu}, t^k)\right)_m.$$

This concludes the proof of the interpolation property (2.11).

Concerning the computational complexity, we see that (i) requires $\mathcal{O}(N\mathrm{card}(S_M))$ operations, (ii) grows as $\mathcal{O}(\mathrm{card}(I_M)J_E^\alpha)$, and (iii) has complexity $\mathcal{O}(M\mathrm{card}(I_M))$. Due to the definition of I_M and the assumption of a local basis, we can upper bound $\mathrm{card}(I_M) \leq MJ$ and $\mathrm{card}(S_M) \leq \mathrm{card}(I_M)J_E = MJJ_E$. Overall, we therefore obtain a complexity estimate for all three steps of $\mathcal{O}(NMJJ_E + MJJ_E^\alpha + M^2J)$, which is linear in N and J, quadratic in M and polynomial in J_E. In particular, the complexity is independent of H. ☐

We indeed obtain H-independent complexity for the complete online stage, in particular for the empirical interpolation of a localized operator evaluation. Therefore the method is suitable for the online-phase in RB methods.

We want to comment on some implementation issues, which make our approach distinct from existing RB approaches. The first comment addresses the fact, that the online-phase is tightly connected to the numerical environment producing the detailed simulations. The reason is that the local functionals $l_i(u)$ must be evaluated, which usually is much more complex than a simple operation of a scalar function operating on u as in [2]. The functionals are operating on discrete functions and therefore require knowledge of the geometry, the numerical grid, neighborhood between cells, data functions, etc. These numerical structures must be available during the reduced simulation. This directly leads to an implementation issue related to the numerical grid. As mentioned earlier, the main nontrivial requirement for complete H-independent computation in the online-phase is indeed depending on a fast evaluation of the coefficient functionals l_i independent of H. Therefore, the grid structure

ETNA
Kent State University
etna@mcs.kent.edu

must allow the selection of subgrids, i.e., access to a part of the grid and access-complexities that are independent of H. The last implementational comment is important for the coefficient functionals l_i: usually these are evaluated simultaneously for all DOFs, i.e., producing the new values from the given ones. In the online-phase however, this evaluation must be limited to a local evaluation, i.e., working on the subgrid, involving all DOFs of the input function corresponding to the degrees of freedom in S_M, but only producing values for the target DOF-indices I_M.

Note that the offline-online decomposition as presented here, can be easily extended such that the online-phase not only allows the choice of a parameter $\boldsymbol{\mu}$, but also the choice of an $N \in \{1, \ldots, N_{max}\}$ and an $M \in \{1, \ldots, M_{max}\}$ for some large $N_{max}, M_{max} \in \mathbb{N}$. The advantage of this is interactive choice of approximation accuracy.

4. A-posteriori error estimate. As a first analytical backup for the presented scheme, we derive an a-posteriori L^2-error estimate. The bound is effectively computable in complexity polynomial in N and M during the reduced simulation. This is due to the fact, that the crucial ingredients in the bound are based on the residual $R^k := (u_N^k - u_N^{k+1} - \Delta t \mathcal{I}_M[\mathcal{L}_E[u_N^k]])/\Delta t$. Its L^2-norm can be computed as

$$
\begin{aligned}
\Delta t^2 \left\|R^k\right\|_{L^2(\Omega)}^2 &= \left\|u_N^k - u_N^{k+1} - \Delta t \mathcal{I}_M[\mathcal{L}_E[u_N^k]]\right\|_{L^2(\Omega)}^2 \\
&= \left\| \sum_{i=1}^{N} (a_i^k - a_i^{k+1})\varphi_i - \Delta t \sum_{m=1}^{M} \mathcal{I}_M[\mathcal{L}_E[u_N^k]](x_m)\xi_m \right\|_{L^2(\Omega)}^2 \\
&= \left\|\mathbf{a}^k - \mathbf{a}^{k+1}\right\|^2 - 2\Delta t(\mathbf{a}^k - \mathbf{a}^{k+1})^T \mathbf{C}_E \mathbf{l}_E[\mathbf{a}^k] + \Delta t^2 (\mathbf{l}_E[\mathbf{a}^k])^T \mathbf{M} \mathbf{l}_E[\mathbf{a}^k], \qquad (4.1)
\end{aligned}
$$

with vectors and matrices from the RB simulation scheme and the mass-matrix $\mathbf{M} \in \mathbb{R}^{M\times M}$ of the interpolation basis given as $(\mathbf{M})_{m,m'} = (\xi_m, \xi_{m'})$. Additionally, in the following estimate we use an extended interpolation space $\mathcal{W}_{M+1}$ and a corresponding interpolation point x_{M+1} obtained by the collateral reduced basis generation algorithm of Section 3.1.1.

PROPOSITION 4.1 (A-Posteriori L^2-Error Bound). *We assume that for all $\boldsymbol{\mu}, t^k$ the operator $Id - \Delta t \mathcal{L}_E(\boldsymbol{\mu}, t^k)$ is Lipschitz-continuous in L^2 with known Lipschitz-constant C_E, i.e., for all $u, u' \in \mathcal{W}_H$, it holds*

$$
\left\|u - u' - \Delta t(\mathcal{L}_E(\boldsymbol{\mu}, t^k)[u] - \mathcal{L}_E(\boldsymbol{\mu}, t^k)[u'])\right\|_{L^2(\Omega)} \le C_E \left\|u - u'\right\|_{L^2(\Omega)}. \qquad (4.2)
$$

We assume that $\mathcal{L}_E(\boldsymbol{\mu}, t^k)[u_N^k] \in \mathcal{W}_{M+1}$. We require that the reduced basis space contains the projections of the initial data components $P[u_0^q] \in \mathcal{W}_N$ for $q = 1, \ldots, Q_{u_0}$. Then for given $\boldsymbol{\mu}$ the RB evolution error at time t^k can be bounded by

$$
\left\|u_H^k(\boldsymbol{\mu}) - u_N^k(\boldsymbol{\mu})\right\|_{L^2(\Omega))} \le \Delta_{N,M}^k(\boldsymbol{\mu}), \qquad (4.3)
$$

with

$$
\Delta_{N,M}^k(\boldsymbol{\mu}) := \sum_{k'=0}^{k-1} \Delta t C_E^{k-1-k'} \left(|\theta_{M+1}^{k'}(\boldsymbol{\mu})| \left\|q_{M+1}\right\|_{L^2(\Omega)} + \|R^{k'}(\boldsymbol{\mu})\|_{L^2(\Omega)} \right), \qquad (4.4)
$$

and the empirical interpolation error estimator

$$
\theta_{M+1}^{k'}(\boldsymbol{\mu}) = \mathcal{L}_E(\boldsymbol{\mu}, t^{k'})[u_N^{k'}](x_{M+1}) - \mathcal{I}_M[\mathcal{L}_E(\boldsymbol{\mu}, t^{k'})[u_N^{k'}]](x_{M+1}). \qquad (4.5)
$$

In particular, the upper bound $\Delta_{N,M}^k(\boldsymbol{\mu})$ can be effectively computed.

ETNA
Kent State University
etna@mcs.kent.edu

Proof. From the construction of the scheme, we obtain for given $\boldsymbol{\mu}$ and given k (abbreviating $\mathcal{L}_E(\boldsymbol{\mu}, t^k)$ by $\mathcal{L}_E$)

$$u_H^{k+1} = u_H^k - \Delta t \mathcal{L}_E[u_H^k], \tag{4.6}$$

$$u_N^{k+1} = u_N^k - \Delta t \mathcal{I}_M[\mathcal{L}_E[u_N^k]] - \Delta t R^k. \tag{4.7}$$

By forming the difference of the two equations we obtain an evolution equation for the error $e^k := u_H^k - u_N^k$ as follows

$$\begin{aligned} e^{k+1} &= e^k - \Delta t \left(\mathcal{L}_E[u_H^k] - \mathcal{I}_M[\mathcal{L}_E[u_N^k]]\right) + \Delta t R^k \\ &= e^k - \Delta t \left(\mathcal{L}_E[u_H^k] - \mathcal{L}_E[u_N^k]\right) + \Delta t \left(\mathcal{I}_M[\mathcal{L}_E[u_N^k]] - \mathcal{L}_E[u_N^k]\right) + \Delta t R^k. \end{aligned}$$

The interpolated operator evaluation can be written in the non-nodal basis $\mathbf{Q}_M$ expansion as

$$\mathcal{I}_M[\mathcal{L}_E[u_N^k]] = \sum_{m=1}^{M} \sigma_m q_m, \tag{4.8}$$

where the coefficients σ_m are obtained by solving (3.4) for index range $i = 1, \ldots, M$. Due to the assumption, the exact evolution $\mathcal{L}_E[u_N^k]$ is contained in $\mathcal{W}_{M+1}$ and can be written as

$$\mathcal{L}_E[u_N^k] = \sum_{m=1}^{M+1} \theta_m^k q_m. \tag{4.9}$$

We recall that by construction of the functions q_m in the collateral basis construction phase, $q_m(x_{m'}) = 0$ for $m' < m$. Comparing the values of (4.8) and (4.9) in the points x_i for $i = 1, \ldots, M+1$ yields that $\theta_m^k = \sigma_m$ for $m = 1, \ldots, M$ and $\theta_{M+1}^k = \mathcal{L}_E[u_N^k](x_{M+1}) - \sum_{m=1}^{M} \sigma_m q_m(x_{M+1})$. Therefore, we obtain

$$\left\| \Delta t \left(\mathcal{I}_M[\mathcal{L}_E[u_N^k]] - \mathcal{L}_E[u_N^k]\right) \right\|_{L^2(\Omega)} = \Delta t |\theta_{M+1}^k| \, \|q_{M+1}\|_{L^2(\Omega)} .$$

Together with the assumption of the boundedness of the discretization operator (4.2) and the residual norm decomposition (4.1), we arrive at

$$\left\| e^{k+1} \right\|_{L^2(\Omega)} \leq C_E \left\| e^k \right\|_{L^2(\Omega)} + \Delta t (|\theta_{M+1}^k| \, \|q_{M+1}\|_{L^2(\Omega)} + \left\| R^k \right\|_{L^2(\Omega)}).$$

We assume that the initial data components are contained in $\mathcal{W}_N$, therefore $P[u_0(\boldsymbol{\mu})] \in \mathcal{W}_N$ for all $\boldsymbol{\mu}$ with the affine parameter dependence (2.7), hence $e^0 = 0$. Thus we can resolve the recursion of the error evolution and obtain the claimed a-posteriori error bound (4.3) and (4.4). ∎

Note that in absence of an interpolation error ($|\theta_{M+1}^k| = 0$), we reproduce the estimate for the linear and affine parameter dependent case [6]. Here we do not assume linearity of the operator and allow a more general parameter dependence.

We briefly comment on the plausibility of the assumptions: The boundedness of the evolution operator (4.2) is realistic. For instance, if $\mathcal{L}_E$ is linear and coercive and Δt is sufficiently small, then this can even be bounded by a suitable $C_E \leq 1$; see, e.g., [6]. The main restricting and unrealistic assumption is found frequently in empirical interpolation estimates [9, 2], which is the approximation quality of the collateral space $\mathcal{W}_{M+1}$. Requiring $\mathcal{L}_E(\boldsymbol{\mu}, t^k)[u_N^k] \in \mathcal{W}_{M+1}$ is unrealistic, since $\mathcal{L}_E(\boldsymbol{\mu}, t^k)[u_H^k]$, not $\mathcal{L}_E(\boldsymbol{\mu}, t^k)[u_N^k]$, are used in the collateral basis generation procedure. To improve this estimate, extensions similar to [9] are possible. This means that not only $\mathcal{W}_{M+1}$ is used in the estimate, but by involving larger

ETNA
Kent State University
etna@mcs.kent.edu

$\mathcal{W}_{M+M'}$ for $M' > 1$, an extended error estimator can be devised. The last assumption, the condition on the initial data, is trivially satisfied if we include the projections of the initial data components in the reduced basis space.

The error estimator readily allows an offline-online decomposition: In the offline phase the interpolation mass matrix $\mathbf{M}$ in (4.1) must be computed. For the error estimate, the collateral reduced basis must be extended by one further function q_{M+1} and interpolation point x_{M+1}. The norm $\|q_{M+1}\|$ and the values of the interpolation basis functions $q_m(x_{M+1})$ must be stored. The online-phase then evaluates the interpolated operator $\mathcal{I}_M[\mathcal{L}_E[u_N^k]]$ at the point x_{M+1}, computes θ_{M+1}^k, assembles the residual norm (4.1) and the final bound (4.4).

The relevance of a-posteriori error estimates in RB schemes is that they provide a certified quality measure for the reduced simulation. This can be used for example in the offline-stage of basis generation, where the error estimator can be used as an indicator, how well certain regions of the parameter space are resolved with a current RB model [6, 10]. Parameters $\boldsymbol{\mu}$ with large error estimators can then be chosen for basis extension, such that the extended model becomes more accurate on these parameters.

5. Experiments. As a model example, we choose the geometry, the P^{2}DE and the FV discretization from [6] and transform the example to a purely explicit evolution scheme by omiting the diffusion. The resulting equation is a convection equation $\partial_t u(\boldsymbol{\mu}, t) + \nabla \cdot (\mathbf{v}u(\boldsymbol{\mu}, t)) = 0$ in $\Omega \times [0, T]$ with $\Omega = [0, 1 \cdot 10^{-3}] \times [0, 2 \cdot 10^{-4}]$, $T = 0.5$, and a space-dependent precomputed velocity field $\mathbf{v}(\mathbf{x})$ as illustrated in Figure 5.1. The boundary segments are assigned different types: noflow Neumann conditions in Γ_3, Γ_6 at the middle of the top and the bottom, outflow conditions at Γ_5 and Dirichlet-conditions on the remaining segments. We consider initial data $u_0(\mathbf{x}) = \frac{1}{2} c_{\text{init}}(\sin(10000\pi x) + 1)$ with a parameter $c_{\text{init}} \in [0, 1]$ interpolating between homogeneous zero initial data and the full sine-wave. The Dirichlet boundary values are set as $b_{\text{dir}}(\mathbf{x}, t) = \beta\chi_{\Gamma_2} + (1 - \beta)\chi_{\Gamma_4}$, where χ_{Γ_i} denote the indicator functions of the corresponding boundary segments. Thus, b_{dir} is parametrized by $\beta \in [0, 1]$, which models concentration differences between the inlet Γ_2 and outlet Γ_4. The Neumann boundary values are chosen as $b_{\text{neu}} = \chi_{\Gamma_5}(\mathbf{v}u) \cdot \mathbf{n}$. The space discretization is a cartesian grid of 40×200 cells, the time range is $t \in [0, T = 0.5]$ discretized with $K = 200$ equally sized time-intervals.

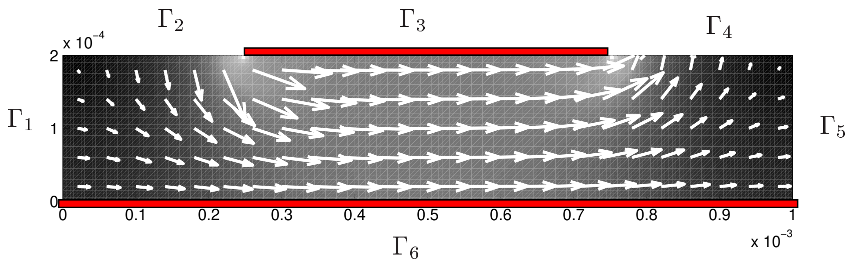

FIGURE 5.1. *Illustration of the geometry and velocity field.*

By this we have specified our P^{2}DE with parameter vector $\boldsymbol{\mu} = (c_{\text{init}}, \beta)^T$ being variable in the range $\mathcal{P} := [0, 1] \times [0, 1]$. We choose a first order explicit finite volume scheme with Lax-Friedrichs-flux for the discretization. A resulting detailed solution for $c_{\text{init}} = 1, \beta = 0$ is illustrated at start- and end-time in Figure 5.2 a) and b). Lowering c_{init} diminishes the sinusoidal data, enlarging β increases the Γ_2 Dirichlet value and lowers the Γ_4 value. For details concerning the numerical scheme and the model example we refer to [6].

As concluded in that study, the case without diffusivity is to some extent a hard case for parametrized model reduction. First, the solution variety is larger, as the smoothing diffusivity

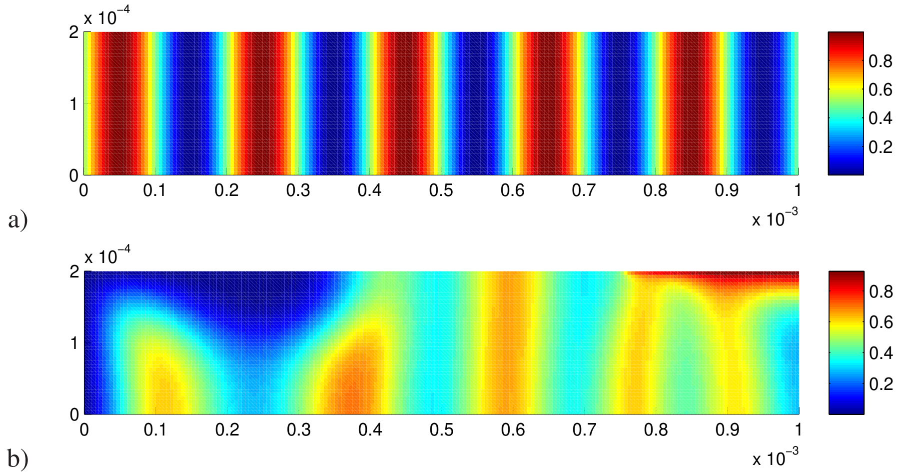

FIGURE 5.2. *Illustration of numerical solutions* $u_H^k(\boldsymbol{\mu})$ *for* $\boldsymbol{\mu} = (c_{\mathrm{init}}, \beta)^T = (1,0)^T$ *at a) start time* $k = 0$*, and at b) end time* $k = 200$*.*

is missing. Secondly, the purely explicit time-step only requires $\mathcal{O}(H) - \mathcal{O}(H^2)$ operations in contrast to $\mathcal{O}(H^3)$ for matrix inversions for parabolic or elliptic problems. Still, we will be able to demonstrate the computational gain. This linear convection problem is inherently affine in u due to the non-homogeneous boundary conditions. This is a good benchmark problem to demonstrate the applicability of the operator interpolation.

In this section we will first demonstrate the results of the empirical interpolation method, then the approximation quality of the proposed RB scheme, and finally the runtime gain of the reduced over the detailed simulation.

5.1. Empirical interpolation. We constructed the collateral reduced basis space $\mathcal{W}_M$ with nodal interpolation basis $\boldsymbol{\xi}_M$ and interpolation points T_M as described in Section 3.1.1, setting $M = 100$ and $M_{train} = \{(\frac{i}{4}, \frac{j}{4}) | i,j = 0,\dots,4\} \subset \mathcal{P}$.

Insights into the interpolation process are obtained from the distribution of the selected interpolation snapshots in the parameter-time space $\mathcal{P} \times [0,T]$, which we plot in Figure 5.3 a). The first observation is, that most interpolation points are almost exclusively gathered at the edges corresponding to the corners of $\mathcal{P}$, i.e., the extreme values of the parameters. This is in accordance with the intuition, that due to the simple parameter dependence, these extreme values produce the most characteristic solutions. Hence, the empirical interpolation automatically detected, that the coarse 5×5 grid of parameter space sampling actially was too fine. A further observation is that the edge corresponding to $\boldsymbol{\mu} = (0,0)^T$ is resolved with only few snapshots. This is due to the fact, that this trajectory has snapshots, that are zero in most of the domain, do not change much in time and therefore are already approximated well by few basis functions. The last observation is that the time-sampling is very dense and more concentrated at early times. This may be due to the fact that the numerical flux has a considerable numerical viscosity, which smoothens the solution. This results in smaller L^2-differences between subsequent snapshots at later times.

A further interesting quantity produced in the offline phase is the distribution of the interpolation points T_M in the computational domain. In our case of piecewise constant functions, the set X_H for selecting the interpolation points is chosen as the cell-centroids. Therefore, we plot the grid-cells corresponding to the selected interpolation points T_M in Figure 5.3 b). Comparing with Figure 5.2 we, indeed, see that the impirical interpolation process selects

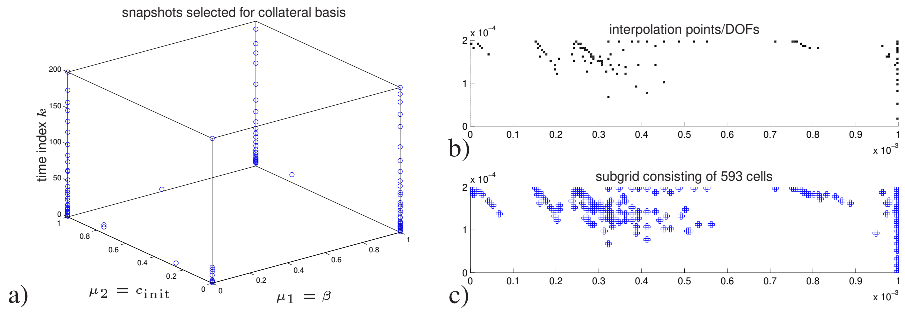

FIGURE 5.3. *Illustration of empirical interpolation offline-quantities. a) Selected empirical-interpolation snapshots in the parameter-time domain $\mathcal{P} \times [0, T]$ with time index k growing in vertical direction, b) grid cells that contain interpolation points T_M ($\equiv$ DOF-index-set I_M), c) subgrid that is extracted and used in the online stage ($\equiv$ DOF-index-set S_M).*

interpolation points that are discriminative for the evolution process. In particular, regions with large gradients are important, as they occur at discontinuities, in our case at the upper Dirichlet-boundaries. This importance is reflected in the higher density of interpolation points in these regions. In case of piecewise constant finite volume spaces, the DOFs can also be identified with grid-cells, so the marked cells in plot b) particularly represent the set I_M of DOFs that are to be computed by every online-step during the empirical interpolation. In plot c) we plot the larger DOF-index set S_M, which is the set of DOFs, that must be available to perform the local evaluation of the evolution operator. For these online-computations also the geometry of the cells must be available. Thus, the marked cells are exactly the subgrid, that is extracted from the detailed grid, and used in the online evaluation of the localized operator. We see, that these subsets of elements are very small compared to the global grid (593 of 8000 elements), guaranteeing efficient online-evaluation.

We now investigate quantitative aspects of the empirical interpolation. A natural measure for the quality of this is the criterion used in the construction of the collateral basis,

$$\max_{v \in L_{train}} \min_{v^* \in \mathrm{span}\{q_i\}_{i=1}^m} \|v - v^*\|_{L^2(\Omega)},$$

for increasing $m = 1, \ldots, M$. This error measures the maximum L^2-projection error over the training set. Additionally, the maximum interpolation error is an interesting quantity, since this is the real error resulting during the interpolation. In Figure 5.4 a) we plot the maximal L^2-projection error over the training set of operator-evaluation-snapshots L_{train} and the maximal interpolation error for increasing dimensionality of the interpolating space for $m = 1, \ldots, M$.

The exponential error decrease in the curves is obvious. Hence, indeed, by minimizing the approximation error over the training set of operator evaluations, the interpolation error is also kept small. In the current simple example, this training error is a very reliable predictor for errors on previously unseen parameters. The diagrams for independent test-sets are almost identical. The test-errors are even frequently smaller, i.e., the training set seems to contain the most difficult parameters in our simple example.

5.2. RB error convergence. After the empirical interpolation, we construct a reduced basis Φ_N for $N = 50$ based on a greedy search over the solution-trajectories of the same set M_{train} as used in the empirical interpolation step. We now assess the error convergence of the final reduced basis scheme, i.e., considering the $L^\infty([0, T], L^2(\Omega))$ error between the detailed and the reduced simulation. We vary several values of N and M and for each resulting RB

ETNA
Kent State University
etna@mcs.kent.edu

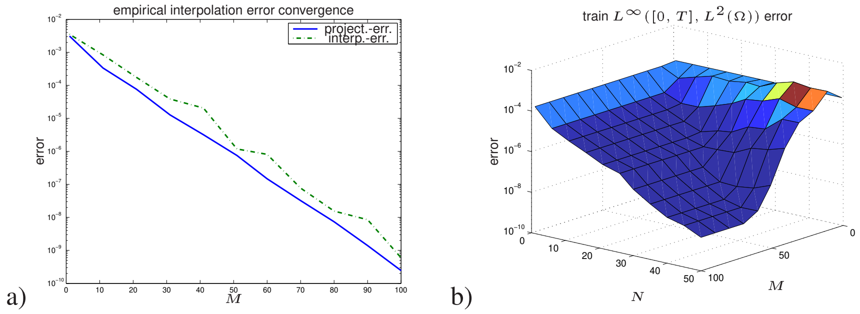

FIGURE 5.4. *Error convergence of empirical interpolation and the resulting RB scheme. a) Decrease of maximum projection and interpolation error with increasing dimensionality of the interpolation space $\mathcal{W}_M$ in the offline phase. b) Convergence of the overall RB scheme, where the maximum error $\|u_H - u_N\|_{L^\infty([0,T],L^2)}$ over M_{train} is plotted for varying values N and M.*

scheme determine the maximum error over the training set M_{train}

$$\max_{\boldsymbol{\mu}\in M_{train}} \|u_N(\boldsymbol{\mu}) - u_H(\boldsymbol{\mu})\|_{L^\infty([0,T],L^2(\Omega))}\,.$$

The resulting errors are depicted in Figure 5.4 b). The results indicate that it is useful to require a certain minimal and maximal ratio of N/M. If N is chosen too large with respect to M, then large errors occur due to the (relatively) poor approximation of the discretization operator. If M is taken too large with respect to N, then the approximation error remains almost constant, so too large M is possible, but a waste of computational time. Similar investigations of the test-error reveal, that the error surfaces are almost identical, which again indicates that our coarse choice of M_{train} is sufficient. The necessary balancing of N and M can also be concluded from theoretical considerations: Let $u_{H,M}$ denote the detailed simulation using the interpolated instead of the exact evolution operator, i.e., $u^0_{H,M} := u^0_H$ and $u^{k+1}_{H,M} = u^k_{H,M} - \Delta t\mathcal{I}_M[\mathcal{L}_E[u^k_{H,M}]]$ for $k = 1,\ldots,K-1$. Then the overall RB approximation error can be decomposed in an empirical interpolation component and a Galerkin-projection component:

$$\left\|u^k_H - u^k_N\right\|_{L^2(\Omega)} \le \left\|u^k_H - u^k_{H,M}\right\|_{L^2(\Omega)} + \left\|u^k_{H,M} - u^k_N\right\|_{L^2(\Omega)}\,.$$

The first term is determined solely by M, for fixed M the second term is mainly depending on N. The regions in the N,M-plane, where either the first or the second term is dominating is nicely reflected in the diagram.

5.3. Computational gain. The main goal of RB approaches is an accelerated online-phase compared to the full simulation. Based on a MATLAB implementation run on an IBM Lenovo Notebook (Intel Centrino Duo, 2.0 GHz, 1024 MB RAM), we obtain the time measurements as given in Tab. 5.1. We compute the averaged runtimes for a detailed simulation and reduced simulations for varying choices of N and M with fixed ratio. The mean runtimes are determined from 10 single simulations. The detailed simulation with full evaluation of the explicit operator in each timestep requires 26.65 seconds, whereas the reduced simulations are computed in 2.83 to 4.22 seconds. For visually indiscriminable solutions, the choice $N = 20, M = 30$ is sufficient, which gives speedup of a factor 8.5 in our case. Recall from an earlier comment that this acceleration will be more pronounced in combination with implicit discretization components, where the operation count for a single step grows with $\mathcal{O}(H^3)$ instead of $\mathcal{O}(H)$ as in our case of localized explicit evolution operators.

TABLE 5.1
Runtime Comparison of detailed simulation and online phase of reduced basis simulations for different approximation levels N, M. The mean over 10 simulations is reported.

Simulation	Approximation	Mean Runtime [s]
detailed	$H = 8000$	26.65
reduced	$N = 10, M = 15$	2.83
reduced	$N = 20, M = 30$	3.15
reduced	$N = 30, M = 45$	3.56
reduced	$N = 40, M = 60$	3.86
reduced	$N = 50, M = 75$	4.22

The gain of the RB approach will be obtained in application settings where the online time complexity is crucial irrespective of a possibly expensive offline-phase. But also in applications, where the cost for the offline-phase must remain decent, RB approaches can be beneficial, if it is a multi-query setting with sufficient number of requests: In our example, the runtimes of the offline-phase are about 60 minutes for construction of $\mathcal{W}_M$ and $\mathcal{W}_N$. For $N = 20$, $M = 30$, we save 23s for each online simulation compared to the detailed model. Hence, after roughly 150 simulation runs with different parameters, the offline-phase pays off.

The results indicate that the reduced model indeed is so fast that it can be applied in an interactive setting. We realized this by incorporating the reduced simulation in an interactive MATLAB-GUI, which allows online-parameter variation by the user.

6. Conclusion. We have presented a reduced basis method for evolution schemes, which have a localized explicit discretization operator. As main ingredient, the empirical interpolation method was adopted to the interpolation of discretization operator evaluations. This required an extensive offline-phase for constructing a collateral reduced basis space, an interpolation scheme based on a subgrid of the detailed grid, and an online reduced simulation scheme. We derived an a-posteriori error estimator with certain restrictions. On a simple model example we have demonstrated the applicability of the RB method. We obtained a runtime gain of factor 6-10 in the reduced model, which allows parameter variation without visible degradation of the solution over the parameter domain. Hereby we demonstrated that RB methods are not only useful in implicit discretizations of evolution problems, as done so far, but also in the more time critical case of explicit discretizations. This speedup is expected to be more pronounced in presence of implicit discretization contributions and higher order time-integration schemes. A further perspective is the application to nonlinear evolution schemes. As we did not explicitly assume linearity of the evolution operator, the current method will be the crucial ingredient for treating the nonlinear case. Examples of such operators are FV schemes or LDG schemes of higher order in space (reconstruction steps, limiters). Further numerical analysis aspects also seem interesting. On one hand this comprises stability statements of the empirical interpolation and the reduced scheme. On the other hand, more general a-posteriori error estimates would be required for certified approximation statements of the reduced simulation.

Acknowledgement. The first author was funded by the German Federal Ministry of Education and Research under grant number 03SF0310C and by the Landesstiftung Baden-Württemberg GmbH. We acknowledge further funding by the Singapore-MIT Alliance and the Progetto Roberto Rocca Politecnico di Milano-MIT. We thank Prof. A.T. Patera for fruitful discussions on the subject, as most of the work was done during a research visit with his group.

REFERENCES

[1] M. BARRAULT, Y. MADAY, N. NGUYEN, AND A. PATERA, *An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations*, C. R. Math. Acad. Sci. Paris Series I, 339 (2004), pp. 667–672.

[2] M. GREPL, *Reduced-basis Approximations and a Posteriori Error Estimation for Parabolic Partial Differential Equations*, Ph.D. thesis, Department of Mechanical Engineering, Massachusetts Institute of Technology, May 2005.

[3] M. GREPL, Y. MADAY, N. NGUYEN, AND A. PATERA, *Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations*, M2AN Math. Model. Numer. Anal., 41 (2007), pp. 575–605.

[4] M. GREPL AND A. PATERA, *A posteriori error bounds for reduced-basis approximations of parametrized parabolic partial differential equations*, M2AN Math. Model. Numer. Anal., 39 (2005), pp. 157–181.

[5] B. HAASDONK AND M. OHLBERGER, *Adaptive basis enrichment for the reduced basis method applied to finite volume schemes*, in Finite Volumes for Complex Applications V, Proceedings of the 5th International Symposium on Finite Volumes for Complex Applications, R. Eymard and J.-M. Hérard, eds., Wiley, New York, 2008, pp. 471–478.

[6] ———, *Reduced basis method for finite volume approximations of parametrized linear evolution equations*, M2AN, Math. Model. Numer. Anal., 42 (2008), pp. 277–302.

[7] D. KRÖNER, *Numerical Schemes for Conservation Laws*, John Wiley & Sons, New York, 1997.

[8] A. LØVGREN, *Reduced Basis Modeling of Hierarchical Flow Systems*, Ph.D. thesis, Department of Mathematical Sciences, Norwegian University of Science and Technology, Trondheim, 2005.

[9] N. NGUYEN, *Reduced-Basis Approximation and A Posteriori Error Bounds for Nonaffine and Nonlinear Partial Differential Equations: Application to Inverse Analysis*, Ph.D. thesis, HPCES Programme, Singapore-MIT Alliance, National University of Singapore, 2005.

[10] A. PATERA AND G. ROZZA, *Reduced Basis Approximation and a Posteriori Error Estimation for Parametrized Partial Differential Equations*, MIT, 2007. Version 1.0, Copyright MIT 2006-2007, to appear in (tentative title) MIT Pappalardo Graduate Monographs in Mechanical Engineering.

[11] D. ROVAS, L. MACHIELS, AND Y. MADAY, *Reduced basis output bound methods for parabolic problems*, IMA J. Numer. Anal., 26 (2006), pp. 423–445.

[12] G. ROZZA, *Shape design by optimal flow control and reduced basis techniques: Applications to bypass configurations in haemodynamics*, Ph.D. thesis, Institut d′Analyse et Calcul Scientifique, École Polytechnique Fédérale de Lausanne, November 2005.

[13] G. ROZZA, D. HUYNH, AND A. PATERA, *Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations*, Arch. Comput. Methods Engrg., 15 (2008), pp. 229–275.

[14] T. TONN AND K. URBAN, *A reduced-basis method for solving parameter-dependent convection-diffusion problems around rigid bodies*, in ECCOMAS CFD 2006 Proceedings, P. Wesseling, E. Onate, and J. Periaux, eds., Delft University of Technology, The Netherlands, 2006.