A Dimensional Reduction Approach Based on the Application of Reduced Basis Methods in the Framework of Hierarchical Model Reduction

The MIT Faculty has made this article openly available. Please share how this access benefits you. Your story matters.

Citation

As Published
http://dx.doi.org/10.1137/130939122

Publisher
Society for Industrial and Applied Mathematics

Version
Final published version

Accessed
Thu Dec 27 06:42:23 EST 2018

Citable Link
http://hdl.handle.net/1721.1/88243

Terms of Use
Article is made available in accordance with the publisher’s policy and may be subject to US copyright law. Please refer to the publisher’s site for terms of use.

Detailed Terms

The MIT Faculty has made this article openly available. Please share how this access benefits you. Your story matters.
A DIMENSIONAL REDUCTION APPROACH BASED ON THE APPLICATION OF REDUCED BASIS METHODS IN THE FRAMEWORK OF HIERARCHICAL MODEL REDUCTION

MARIO OHILBERGER† AND KATHRIN SMETANA‡

Abstract. In this article we introduce a new dimensional reduction approach which is based on the application of reduced basis (RB) techniques in the hierarchical model reduction (HMR) framework. Considering problems that exhibit a dominant spatial direction, the idea of HMR is to perform a Galerkin projection onto a reduced space, which combines the full solution space in the dominant direction with a reduction space in the transverse direction. The latter is spanned by modal orthonormal basis functions. While so far the basis functions in the HMR approach have been chosen a priori [S. Perotto, A. Ern, and A. Veneziani, Multiscale Model. Simul., 8 (2010), pp. 1102–1127], for instance, as Legendre or trigonometric polynomials, in this work a highly nonlinear approximation is employed for the construction of the reduction space. To this end we first derive a lower dimensional parametrized problem in the transverse direction from the full problem where the parameters reflect the influence from the unknown solution in the dominant direction. Exploiting the good approximation properties of RB methods, we then construct a reduction space by applying a proper orthogonal decomposition to a set of snapshots of the parametrized partial differential equation. For an efficient construction of the snapshot set we apply adaptive refinement in parameter space based on an a posteriori error estimate that is also derived in this article. We introduce our method for general elliptic problems such as advection-diffusion equations in two space dimensions. Numerical experiments demonstrate a fast convergence of the proposed dimensionally reduced approximation to the solution of the full dimensional problem and the computational efficiency of our new adaptive approach.

Key words. dimensional reduction, hierarchical model reduction, reduced basis methods, a posteriori error estimation, adaptive modeling, finite elements

AMS subject classifications. 65N30, 65N15, 65Y20, 35J25

DOI. 10.1137/130939122

1. Introduction. Many phenomena in nature have dominant spatial directions along which the essential dynamics occur. Examples are blood flow problems, fluid dynamics in pipes or river beds, and subsurface flow. This property can be exploited to derive a dimensionally reduced model. However, the processes in the transverse directions are often too relevant for the whole problem to be neglected. This can be due to local features as the presence of a narrowing (stenosis) of an artery or the outlet of a river in a lake, but also to possibly nonlocal processes such as infiltration of soil. To obtain a good approximation of the full dimensional problem, it is hence preferable that the dimensionally reduced model includes information on the transverse dynamics.

In this paper we apply reduced basis (RB) techniques in the hierarchical model reduction (HMR) framework—this yields the HMR-RB approach as briefly outlined in [26]—to approximate partial differential equations (PDEs) in $d$ (space) dimensions...
by a coupled reduced system in \( k \) dimensions, where \( 1 \leq k < d \). The key idea of HMR, originally introduced in [39, 40, 41], is to perform a Galerkin projection of the original variational problem onto a reduced space of approximation order \( m \) of the form

\[
V_m = \{ v_m(x, y) = \sum_{l=1}^{m} \bar{v}_l(x) \phi_l(y), x \in \mathbb{R}^k, y \in \mathbb{R}^{d-k} \},
\]

where the coefficients \( \bar{v}_l(x) \), \( l = 1, \ldots, m \), are defined on the dominant directions and the orthonormal functions \( \phi_l, l = 1, \ldots, m \), on the transverse directions. The projection is thus determined by the choice of the reduction space in the transverse directions \( Y_m := \text{span}(\phi_1, \ldots, \phi_m) \).

For elliptic boundary value problems in thin domains, an optimal a priori choice of reduction spaces was discussed in [39, 40, 41] and a polynomial convergence rate in \( m \), which depends on the regularity of the full solution, is proved for domains of arbitrary size in [40]. In a more general geometrical setting the application and applicability of the HMR approach for problems that exhibit a dominant flow direction has been studied and demonstrated in [16, 30, 32, 33]. Adaptive variants of HMR that allow for a local adaption of the approximation order \( m \) by a domain decomposition scheme [30, 32] or by associating the basis functions to the nodes of a finite element (FE) partition in the dominant direction [33] are introduced. While in all previous papers the choice of the reduction space is based on an a priori chosen set of basis functions such as trigonometric or Legendre polynomials, one of the major contributions of the present work is to employ a double stage (and thus highly) nonlinear approximation in the sense of [13] to construct the reduction space. First, we derive a parametrized lower dimensional problem in the transverse direction from the full problem, where the parameters reflect the influence from the unknown solution in the dominant direction. For the selection of the basis \( \{ \phi_l \}_{l=1}^{m} \) from the solution manifold formed by the solutions of the parametrized dimensionally reduced problem, we apply RB methods, exploiting their good, and in some cases even optimal, approximation properties [5, 7, 14, 22, 34]. In particular we make use of an adaptive training set extension in the parameter space, which combines ideas from [20, 21] and [15] for an efficient generation of a snapshot set. One central ingredient of this construction process is a reliable and efficient a posteriori error estimator for the discretized reduced model which is also derived. On the snapshot set a proper orthogonal decomposition (POD) is applied to construct the reduction space \( Y_m \), where the approximation order \( m \) is determined automatically by the POD by prescribing a certain error tolerance. Thus, both in the construction of the solution manifold and the subsequent choice of the basis functions, information on the full solution is included to obtain a fast convergence of the reduced solution to the full one.

There is a large variety of approaches in the fields of dimensional reduction and tensor based approximations. Thus, we only outline the ones most closely related to our method. One of the earliest techniques is the asymptotic expansion which has among others been applied to derive lower dimensional models for plates (see, e.g., [12]), water waves (see, e.g., [38]), or groundwater flow (see, e.g., [3]). The method of asymptotic partial domain decomposition for thin domains (cf. [17, 27]) generalizes the method of asymptotic expansion. Here, in the (small) part of the domain, where an asymptotically singular behavior of the solution is expected, the full problem is considered, while in the rest of the domain an asymptotic expansion is applied, which is also used to identify suitable coupling conditions at the interface. The main drawback of methods based on asymptotic expansion is that they are only valid if the considered domain is very thin, or, equivalently, the solution is constant along the transverse direction. HMR overcomes this difficulty and constitutes a hierarchical approach to interpolate between such lower dimensional models, derived with a projection approach as presented above with approximation order \( m = 1 \), and their full
dimensional counterparts. A more restricted way of such interpolation is given by the geometrical multiscale approach, where, based on a domain decomposition, the dimension of the model is adjusted locally in an a priori manner (cf. [18, 19] and references therein) and in an adaptive way using a posteriori error information in [25]. Another example for intermediate models are multilayer shallow water systems (cf. [2, 37]) where the flow domain is discretized in the transverse direction by introducing intermediate water heights. Then, in each layer, a classical shallow water equation is considered and a certain coupling between the layers is introduced. Finally, the proper generalized decomposition (PGD) approach (cf. [1, 8, 11, 23] and references therein) also employs a truncated tensor product decomposition for the approximation of the solution but determines the tensor products of the expansion by iteratively solving the Euler–Lagrange equations associated to the considered problem. In contrast to RB methods and the HMR-RB approach proposed in this article, PGD aims at constructing an approximation based on the knowledge of the considered differential operator and not on a priori knowledge on the solution or an approximation of it.

The article is organized as follows. In the next section we introduce a specific problem setting and recall the HMR approach following the framework presented in [32]. The main new contributions of this article are developed in section 3 where we first derive a suitable parametrized equation in transverse direction and then detail the usage of RB techniques to construct the reduction space. In section 4 we derive an a posteriori error estimator that is used in the construction process of the reduction space. Subsequently, we thoroughly discuss the complexity of the resulting HMR-RB approach in section 5. Finally, we present several numerical experiments in section 6 to validate the approximation properties and the computational efficiency of our approach and draw some conclusions in section 7.

2. HMR for elliptic boundary value problems. Let \( \Omega \subset \mathbb{R}^2 \) be a computational domain with Lipschitz boundary \( \partial \Omega \). We define the solution space \( V \) such that \( H^1(\Omega) \subseteq V \subseteq H^1(\Omega) \) and consider the following general elliptic problem:

\[
\text{Find } p \in V : \quad a(p, v) = f(v) \quad \forall v \in V,
\]

where \( a(\cdot, \cdot) \) is a coercive and continuous bilinear form and \( f \) a linear form. Furthermore, we denote \( a_s(\cdot, \cdot) \) the symmetric part of \( a(\cdot, \cdot) \), i.e., for all \( v, w \in V \) there holds \( a_s(v, w) = \frac{1}{2}(a(v, w) + a(w, v)) \). We define a \( V \)-inner product and the induced \( V \)-norm as \((\cdot, \cdot)_V := a_s(\cdot, \cdot)\) and \( \| \cdot \|_V := \sqrt{(\cdot, \cdot)_V} \). Finally, we define the coercivity and the continuity constants of the bilinear form \( a(\cdot, \cdot) \) with respect to the \( V \)-norm as \( c_0 := \inf_{v \in V} (a(v, v)/\|v\|_V^2) \) and \( c_1 := \sup_{v \in V} \sup_{w \in V} (a(v, w)/\|v\|_V\|w\|_V) \). We adopt the HMR framework introduced by Perotto, Ern, and Veneziani [16, 32]. Thus, we refer to (2.1) as the full problem and assume that \( \Omega \) can be considered as a two-dimensional (2D) fiber bundle

\[
\Omega = \bigcup_{x \in \Omega_{1D}} \{x\} \times \omega_x,
\]

where \( \Omega_{1D} \) is the one-dimensional (1D) computational domain in the dominant direction and \( \omega_x \) the transverse fiber associated with \( x \in \Omega_{1D} \). For the sake of simplicity we assume \( \Omega_{1D} \) to be a straight line, that is, \( \Omega_{1D} = [x_0, x_1] \) and denote

\[
\Gamma_0 = \{x_0\} \times \omega_{x_0}, \quad \Gamma_1 = \{x_1\} \times \omega_{x_1}, \quad \Gamma_* = \bigcup_{x \in \Omega_{1D}} \{x\} \times \partial \omega_x.
\]
Furthermore, we define for any \( x \in \Omega_{1D} \) the mapping \( \psi(\cdot; x) : \omega_x \to \hat{\omega} \) between the fiber \( \omega_x \) associated with \( x \in \Omega_{1D} \) and a reference fiber \( \hat{\omega} \) with \( \hat{\omega} = ]y_0, y_1[ \). We adopt the notation for \( z = (x, y) \) being a generic point in \( \Omega \) and \( \hat{z} = (x, \hat{y}) \) being the corresponding point in \( \hat{\Omega} \). The latter is constructed via the mapping \( \Psi : \Omega \to \hat{\Omega} \) depicted in Figure 1 with \( \hat{y} = \psi(y; x) \) for \( y \in \omega_x \) and \( \hat{x} = x \) for \( x \in \Omega_{1D} \), which is why \( \Omega_{1D} \equiv \Omega_{1D} \). We suppose that \( \psi(\cdot; x) \) is a \( C^1 \)-diffeomorphism and that the transformation \( \Psi \) is differentiable with respect to \( z \).

For the formulation of the reduced problem we exploit the fiber structure of \( \Omega \) to define the spaces \( X \) and \( Y \) such that \( H_0^1(\Omega_{1D}) \subseteq X \subseteq H^1(\Omega_{1D}) \) and \( H_0^1(\hat{\omega}) \subseteq Y \subseteq H^1(\hat{\omega}) \). In addition, both spaces have to be compatible with the prescribed boundary conditions on \( \partial \Omega \). On \( \hat{\omega} \) we introduce a set of basis functions \( \{ \phi_k \}_{k \in \mathbb{N}} \in Y \), orthonormal with respect to the \( L^2 \)-inner product on \( \hat{\omega} \), i.e., \( \int_{\hat{\omega}} \phi_k(\hat{y}) \phi_l(\hat{y}) d\hat{y} = \delta_{kl} \) for all \( k, l \in \mathbb{N} \), where \( \delta_{kl} \) is the Kronecker symbol. In previous papers, possible choices for \( \{ \phi_k \}_{k \in \mathbb{N}} \) were suitable a priori chosen functions, like trigonometric \([16,32]\) or Legendre polynomials \([32]\). A major new contribution of this article is to replace the a priori chosen basis by a posteriori constructed basis functions that are tailored to the specific problem at hand (cf. section 3). By combining the space \( X \) with the reduction space \( Y_m = \text{span}(\phi_1, \ldots, \phi_m) \), we define the reduced space

\[
V_m = \left\{ v_m(x, y) = \sum_{k=1}^m \overline{v}_k(x) \phi_k(\psi(y; x)), \quad \overline{v}_k(x) \in X, x \in \Omega_{1D}, y \in \omega_x \right\},
\]

where \( m \in \mathbb{N} \) is the approximation order and the coefficients satisfy

\[
\overline{v}_k(x) = \int_{\omega} v_m(x, \psi^{-1}(\hat{y}; x)) \phi_k(\hat{y}) d\hat{y}, \quad k = 1, \ldots, m.
\]

A Galerkin projection onto \( V_m \) yields the reduced problem for \( p_m \in V_m \) with \( p_m(x, y) = \sum_{k=1}^m \overline{v}_k(x) \phi_k(\psi(y; x)) \) such that

\[
a(p_m, v_m) = f(v_m) \quad \forall \, v_m \in V_m.
\]

For the computation of the coefficient functions \( \overline{v}_l(x) \), \( l = 1, \ldots, m \), we introduce a subdivision \( T_H \) of \( \Omega_{1D} \) with elements \( T_i = (x_{i-1}, x_i) \) of width \( H_i = x_i - x_{i-1} \) and maximal step size \( H := \max_i H_i \). We also introduce an associated conforming FE space \( X^H \subset X \) with \( \text{dim}(X^H) = N_H < \infty \) and basis \( \xi^H_i, i = 1, \ldots, N_H \). \( X^H \) is then combined with \( Y_m \) to define the discrete reduced space

\[
V_m^H = \left\{ v_m^H(x, y) = \sum_{k=1}^m \overline{v}_k^H(x) \phi_k(\psi(y; x)), \quad \overline{v}_k^H(x) \in X^H, x \in \Omega_{1D}, y \in \omega_x \right\}.
\]
By rewriting the discrete reduced solution $p_h^M(x, y)$ as $p_h^M(x, y) = \sum_{k=1}^m \bar{p}_k^H(x) \phi_k(y; x)$ with $\bar{p}_k^H(x) = \sum_{j=1}^{N_H} p_{k,j}^H(x)$ for some coefficients $p_{k,j}^H \in \mathbb{R}$ we obtain the discrete reduced problem: Find $\bar{p}_{k,j}^H \in \mathbb{R}$, $k = 1, \ldots, m$, $j = 1, \ldots, N_H$, such that

$$
\sum_{k=1}^m \sum_{j=1}^{N_H} a(\bar{p}_{k,j}^H \xi_k^H \phi_j, \xi_l^H \phi_l) = f(\xi_l^H \phi_l) \quad \text{for} \quad i = 1, \ldots, N_H \quad \text{and} \quad l = 1, \ldots, m.
$$

Note that for a given set of basis functions $\{\phi_k\}_{k=1}^m$ the integrals in the transverse direction can be precomputed. The computation of $p_h^M(x, y)$ thus reduces to the solution of a 1D coupled system of size $m \times N_H$.

### 3. HMR-RB approach.

The goal of this section is to construct a low dimensional reduction space $Y_m$ which approximates well the transverse behavior of the full problem (2.1) and can be used to set up the approximation spaces $V_m$ (2.3) and $V_h^M$ (2.5) in the HMR framework. Starting from the full problem we derive initially a 1D PDE in transverse direction in which the unknown behavior in the dominant direction enter as parameters (section 3.1). The FE solutions of the corresponding parameter dependent discrete 1D problem (section 3.1) form a solution manifold $M^h$ (cf. (3.7) below), which can be approximated by RB methods with very few basis functions, at least if the manifold is smooth [5, 7, 14, 22, 34]. The key idea is to exploit the good approximation properties of RB methods for the construction of a low dimensional reduction space $Y_m^h$, where the index $h$ indicates that $Y_m^h$ contains discrete solutions. Precisely, we first use an adaptive training set extension similar to the one introduced in [20, 21] for an efficient generation of a snapshot set $M_{0,2}^h \subset M^h$ and subsequently apply a POD to find the principal components of $M_{0,2}^h$, which then form the reduction space $Y_m^h$ of dimension $m$ (cf. section 3.2).

#### 3.1. Derivation of a parametrized 1D problem in transverse direction.

One of the major challenges when deriving a lower dimensional PDE from a full problem is to realize a tensor product decomposition of the full solution. The approach we pursue in this article is to assume (only for the derivation of a suitable 1D PDE in transverse direction) that

$$
p(x, \hat{y}) \approx U(x) \cdot \mathcal{P}(\hat{y}).
$$

Here, the function $U(x)$ represents the behavior of the full solution in the dominant direction, which is unknown at this stage. By choosing the test functions as $v(x, y) = U(x) \cdot \mathcal{P}(\hat{y})$ for any $v \in Y$ we obtain a reduced problem: Given any $U \in X$, find $\mathcal{P} \in Y$ such that

$$
a(U \mathcal{P}, Uv) = f(Uv) \quad \forall v \in Y.
$$

As $U(x)$ is unknown, the integrals in the dominant direction cannot be precomputed. We therefore introduce for an arbitrary integrand $t \in L^1(\hat{\Omega})$ of an integral $I(t) := \int_{\hat{\Omega}} \int_{\Omega_D} t(x, \hat{y}) \, dx \, d\hat{y}$ the quadrature formula

$$
Q(t) := \sum_{l=1}^Q \alpha_l \int_{\hat{\Omega}} \tilde{t}(x_l^q, \hat{y}) \, d\hat{y}, \quad \tilde{t}(x_l^q, \hat{y}) := \lim_{\varepsilon \to 0} \frac{1}{|B_\varepsilon(x_l^q)|} \int_{B_\varepsilon(x_l^q)} t(x, \hat{y}) \, dx,
$$

where $\alpha_l$, $l = 1, \ldots, Q$ are the weights, and $x_l^q$, $l = 1, \ldots, Q$ are the quadrature points. Moreover, $B_\varepsilon(x_l^q) = \{x \in \Omega_D : |x - x_l^q| < \varepsilon\}$ denotes a ball of radius $\varepsilon > 0$ around
we obtain the approximations $a^\delta(\cdot;\cdot)$ and $f^\delta(\cdot)$. The reduced problem with quadrature then reads as follows:

(3.4) Given any $U \in X$, find $P \in Y$: $a^\delta(U P, U v) = f^\delta(U v) \quad \forall v \in Y$.

Next we introduce a parametrization of (3.4) with the objective of applying RB methods (section 3.2) to find optimal locations of the quadrature points and include information on the unknown behavior $U$ of the solution in the dominant direction in the reduction space $Y_m$. For that purpose we define a parameter vector $\mu$, which contains both the quadrature points $x_l^q$, $l = 1, \ldots, Q$, and all evaluations of $U$ and its derivatives in $x$-direction in $x_l^q$, $l = 1, \ldots, Q$, that occur in (3.4), which are $U(x_l^q)$ and $\partial_x U(x_l^q)$, $l = 1, \ldots, Q$. The parameter $\mu$ thus has the form $\mu = (x_1^q, U(x_1^q), \partial_x U(x_1^q), \ldots, x_Q^q, U(x_Q^q), \partial_x U(x_Q^q))$. The evaluations in the interval boundaries of $\Omega_{1D}$ ($x_0$ and $x_1$) have to be added to $\mu$ if nonhomogeneous Neumann or Robin boundary conditions are prescribed in the full problem (2.1) on $\Gamma_0$ or $\Gamma_1$, where $\Gamma_i$, $i = 0, 1$ are defined in (2.2). The parameter space $D$ of dimension $P$, which contains all admissible parameter values of $\mu$, is then defined as $D := [\Omega_{1D} \times I_0 \times I_1]^Q \subset \mathbb{R}^P$. The intervals $I_k \subset \mathbb{R}$, $k = 0, 1$, contain the ranges of $\partial_x U(x)$, $k = 0, 1$. They can be chosen by using a priori information on the solution but might need to be updated iteratively employing a posteriori information gained from reduced approximations. We obtain the following parametrized 1D PDE in the transverse direction:

(3.5) Given any $\mu \in D$, find $P(\mu) \in Y$: $a^\delta(P(\mu), v; \mu) = f^\delta(v; \mu) \quad \forall v \in Y$.

Possible choices for the quadrature formula 3.3 are a modified composite rectangle formula or a standard composite trapezoidal rule (cf. [35]). For the computation of snapshots (i.e., solutions of (3.5) for a given parameter $\mu$), we introduce a subdivision $\tau_h$ of $\tilde{\omega}$ with elements $\tau_j = (y_{j-1}, y_j)$ of width $h_j = y_j - y_{j-1}$ and maximal step size $h := \max_j h_j$. Furthermore, we introduce an associated conforming FE space $Y^h \subset Y$ with $\dim(Y^h) = n_h < \infty$ and basis $v_j^h$, $j = 1, \ldots, n_h$. The parameter dependent discrete 1D problem then reads as follows: Given any $\mu \in D$

(3.6) find $P^h(\mu) \in Y^h$: $a^\delta(P^h, v_j^h; \mu) = f^\delta(v_j^h; \mu)$ for $j = 1, \ldots, n_h$.

### 3.2. Basis generation with RB techniques.

In this subsection we introduce the ADAPTIVE-HMR-RB algorithm to construct the low dimensional reduction space $Y^h_m = \text{span}(\phi_1, \ldots, \phi_m) \subset Y^h$ based on sampling strategies from the RB framework.

We first define the solution manifold $M^h$ and a discrete subset $M^h_{\Xi} \subset M^h$ through

(3.7) $M^h := \{P^h(\mu) | \mu \in D\}$, \quad $M^h_{\Xi} := \{P^h(\mu) | \mu \in \Xi \subset D\}$,

where $P^h(\mu)$ solves (3.6). For an efficient snapshot generation and hence construction of $M^h_{\Xi}$, we use an adaptive training set extension resembling the one introduced in [20, 21]. This adaptive refinement of the parameter space is performed by Algorithm 1 ADAPTIVETRAINEXTENSION, which is described in detail below. Note that different from the standard greedy algorithm, we are interested in finding snapshots that yield a good approximation $p^H_m$ of a certain 2D FE reference solution $p^{H \times h}$ with respect to the $V$-norm. Therefore, we refine the parameter space near the parameter value for which $p^{H \times h}$ is best approximated by $p^H_m$ in order to find snapshots $P^h(\mu)$ which may even yield a better approximation. Finally, we apply a POD (Algorithm 2) to determine the principal components of $M^h_{\Xi}$, which in turn span the reduction space $Y^h_m$, where $m \ll |\Xi|$.
Algorithm 1: Adaptive training set extension and snapshot generation.

\[
\text{ADAPTIVETRAINEXTENSION}(G_0, \Xi_{G_0}, m_{\text{max}}, i_{\text{max}}, n_G, \theta, \sigma_{\text{thres}}, N_H)
\]

\[
\text{Initialize } G = G_0, \Xi = \Xi_{G_0}, \phi_0 = \emptyset, \rho_0(G) = 0
\]

for \(m = 1 : m_{\text{max}}\) do

\[
\text{Compute } \mathcal{P}_G^h
\]

\[
[\eta(G), \sigma(G)] = \text{ElementIndicators}(\{\phi_k\}_{k=1}^{m-1}, \mathcal{P}_G^h, G, \rho(G), N_H')
\]

for \(i = 1 : i_{\text{max}}\) do

\[
G := \text{MARK}(\eta(G), \sigma(G), \theta, \sigma_{\text{thres}})
\]

\[
(G, \Xi_G) := \text{REFINEN}(G, \Xi_g, n_G)
\]

\[
\rho(G \setminus G) = \rho(G \setminus G) + 1
\]

Compute \(\mathcal{P}_G^h
\]

\[
[\eta(G), \rho(G), \sigma(G)] = \text{ElementIndicators}(\{\phi_k\}_{k=1}^{m-1}, \mathcal{P}_G^h, G, N_H')
\]

end

\[
\{\phi_k\}_{k=1}^m := \text{POD}(\mathcal{P}_G^h, m)
\]

end

return \(\mathcal{P}_G^h, \Xi_G\)

Algorithm 1: ADAPTIVETRAINEXTENSION. Let \(G\) denote a hyper-rectangular, possibly nonconforming, adaptively refined grid in the parameter space \(\mathcal{D}\), \(g\) a cell of \(G\) and \(N_G\) the number of cells in \(G\). Different from the approach in [20, 21], the training set \(\Xi_g\) consists of parameter values which are sampled from the uniform distribution over the cell \(g\). \(n_G\) denotes the sample size of \(\Xi_g\) and is chosen identical for all elements. Finally, we define the overall training set \(\Xi_G = \bigcup_{g \in G} \Xi_g\). Inspired by [20, 21] we use a local mesh adaptation with a \text{SOLVE} \rightarrow \text{ESTIMATE} \rightarrow \text{MARK} \rightarrow \text{REFINE} strategy to generate \(G\) and \(\Xi_G\) from a given coarse partition \(G_0\) of the parameter space and an associated initial train sample \(\Xi_{G_0}\). To estimate the error between \(p_m^H\) and \(\eta_h^H \times h\) in the \(V\)-norm, we derive in section 4 a reliable and efficient error estimator \(\Delta_m\). In order to detect the best approximating snapshots \(\mathcal{P}_G^h(\mu)\), we define an element indicator

\[
(3.8) \quad \eta(g) := \min_{\mu \in \Xi_g} \Delta_m(\mu).
\]

Next, we fix \(\theta \in (0, 1)\) and mark in each iteration the \(\theta N_G\) elements with the smallest indicators \(\eta(g)\) for refinement. It is well known in the context of adaptive finite element methods (FEMs) that such an indicator may cause problems if the initial mesh is too coarse to resolve local structures of the data. Thus, we use as in [20, 21] an additional criterion to decide whether an element is marked for refinement or not and define a second indicator

\[
(3.9) \quad \sigma(g) := \text{diam}(g) \cdot \rho(g).
\]

Here, \(\rho(g)\) counts the number of iterations in which the cell \(g\) has not been refined, since its last refinement and \(\text{diam}(g)\) denotes the diameter of \(g\). If \(\sigma(g)\) lies above a certain threshold \(\sigma_{\text{thres}}\), the element \(g\) is marked for refinement as well. This leads asymptotically to a refinement of all elements. All elements marked for refinement are bisected in each direction, leading to \(2^p - 1\) new elements per refined element, where \(P = \text{dim}(\mathcal{D})\). To generate the training sets of the new elements, we first sort \(\Xi_{g_{\text{parent}}}\) into the new elements \(g_{\text{children}}\). Then, in each children \(g_{\text{children}}\), we sample new parameter values from the uniform distribution over \(g_{\text{children}}\) until the sample size of \(\Xi_{g_{\text{children}}}\) reaches \(n_G\). The complete adaptive snapshot generation procedure is
described in Algorithm 1 in an abstract way. During each loop over the model order \( m \), first the snapshots \( P_k^G \) and the element indicators \( \eta(G) \) and \( \sigma(G) \) are computed for all elements in \( G \). Then the parameter space is adaptively refined in order to find the best approximation using \( m \) basis functions. Finally, the \( m \) principal components \( \{ \phi_k \}_{k=1}^m \) of the set of snapshots \( P_k^G \) are identified by a POD. We highlight that throughout Algorithm 1 the computations dependent on the number of degrees of freedom in the dominant direction, namely, the computation of the reduced solution and the error indicator \( \eta(g) \), are not performed in the possibly high dimensional space \( V_m^H \) but in a lower dimensional space \( V_m^{H'} \) of dimension \( N_{H'} \cdot m \), where \( N_{H'} \ll N_H \). This completes the description of Algorithm 1.

As \( G \) is a product-like hyper-rectangular grid, the applicability of Algorithm 1 is limited to small parameter dimensions \( P \). We note that the number of quadrature points in (3.5) and thus the dimension of the parameter space may be limited by applying HMR-RB within a domain decomposition framework. Here, we would assign a training set to each subdomain which would also considerably reduce the size of the training set \( \Xi \). Furthermore, we mention that in contrast to [20, 21], where the vertices of \( G \) generate the training set, we choose \( \Xi \) randomly in order to avoid repetitions, due to parameters lying on the same edge of a cell. We have borrowed the idea to choose the training sets of the cells randomly from [15]. Here a Voronoi tessellation is applied, which is why this approach could be a feasible option to extend the HMR-RB approach to higher parameter dimensions. Alternatively, we may also realize an anisotropic refinement of the parameter space as proposed in [24] by employing a distance function based on the Hessian of the RB approximation for each \( \mu \in \Xi \) or use a clustering algorithm as, for instance, in [29]. The generation of \( \{ \phi_k \}_{k=1}^m \) can also be done by a greedy algorithm which adds in each iteration the basis function belonging to the parameter \( \mu \) with the smallest value of \( \Delta_m \). However, numerical experiments showed much better performance for the POD approach, as more linear independent snapshots are found during the application of Algorithm 1.

**Algorithm 2: Adaptive-HMR-RB.** First, we initialize the train sample \( \Xi \) by sampling \( n_\Xi \) parameter values from the uniform distribution over each cell \( g \in G_0 \). Then Algorithm 1 is called for the efficient generation of the snapshots \( P_k^G \). Setting \( \Xi = \Xi_0 \) and \( n_{\text{train}} = |\Xi| \), we identify the principal components of span\{\( M_{\Xi_0}^h \)\} by a POD, where the POD spaces \( Y_{k}^{\text{POD}}, \ k = 1, \ldots, m, \) are defined as the solutions of the optimization problem

\[
Y_k^{\text{POD}} = \arg \min_{W_k \subseteq \text{span}\{\mathcal{M}_{\Xi_0}^h\}} \left( \frac{1}{n_{\text{train}}} \sum_{\mu \in \Xi} \inf_{w_k \in W_k} \| P^h(\mu) - w_k \|_{L^2(\omega)}^2 \right).
\]

The POD error \( e_k^{\text{POD}}, \ k = 1, \ldots, m, \) is defined as

\[
e_k^{\text{POD}} := \left( \frac{1}{n_{\text{train}}} \sum_{\mu \in \Xi} \inf_{w_k \in Y_k^{\text{POD}}} \| P^h(\mu) - w_k \|_{L^2(\omega)}^2 \right)^{1/2}.
\]

Demanding that \( e_m^{\text{POD}} \leq \epsilon_{\text{tol}} \), where \( \epsilon_{\text{tol}} \) is a prescribed error tolerance, we set \( Y_m^{\text{POD}} := Y_m^{\text{POD}} \). This completes the description of Algorithm 2 (Adaptive-HMR-RB). The choice of the input parameters \( m_{\max}, i_{\max}, n_\Xi, \sigma_{\text{thres}}, \) and \( N_{H'} \) will be discussed in detail in section 6. It is well known that the optimization problem (3.10) is equivalent to the solution of a \( n_{\text{train}} \times n_{\text{train}} \) correlation matrix eigenvalue problem [28, 36]. The POD error (3.11) can then be computed as \( e_m^{\text{POD}} = (\sum_{k=m+1}^{n_{\text{train}}} \lambda_k) / 2 \), where \( \lambda_k \)
Algorithm 2. Construction of the reduction space $Y^h_m$.

Initialize $\Xi_{G_0}$

$[P^h_{G_0}, \Xi_{G_0}] =$ ADAPTIVE-TRAINEXTENSION($G_0, \Xi_{G_0}, m_{\max}, i_{\max}, n_\Xi, \theta, \sigma_{\text{thres}}, N_H', \varepsilon_{\text{tol}}$)

$Y^h_m := \text{POD}(P^h_G, \varepsilon_{\text{tol}})$, such that $e^\text{POD}_{m} \leq \varepsilon_{\text{tol}}$

return $Y^h_m$

are the eigenvalues of the just mentioned eigenvalue problem [28]. We point out that by definition the POD space approximates $\mathcal{M}_\Xi^2$ (3.7). A validation if the selected basis functions approximate the solution $p(x, y)$ of the full problem 2.1 with the same approximation quality as $\mathcal{M}_\Xi^2$ will therefore be performed in section 6.

4. A posteriori error estimation. In this section we derive as in the RB framework [28, 36] a reliable and efficient error estimator for the (model) error between the discrete reduced solution $p^h_m$ and a certain reference solution. To define the latter we introduce a partition $\hat{T} := T_H \times \tau_n$ of $\hat{\Omega}$ induced by the subdivisions $T_H$ of $\Omega_{1D}$ and $\tau_n$ of $\omega$ defined in section 2 and section 3.1. The elements of $\hat{T}$ are defined as $T_{i,j} := T_i \times T_j$, where $T_i \in T_H$ and $T_j \in \tau_n$. Following the notation of [6] we assume that $X^H$ and $Y^h$, defined in section 2 and section 3.1, coincide with Lagrange FE spaces of $k$th and $l$th order and can therefore be defined as $X^H := \{ w^H \in C^0(\Omega_{1D}) : w^H|_{T_i} \in \mathbb{P}_k(T_i), T_i \in T_H \}$ and $Y^h := \{ w^h \in C^0(\hat{\omega}) : w^h|_{T_j} \in \mathbb{P}_l(T_j), T_j \in \tau_n \}$. Here, $\mathbb{P}_k(T_i)$ denotes the set of polynomials of order $\leq k$ over $T_i$ in one variable. We further suppose that $\xi^H, i = 1, \ldots, N_H$ and $\psi^h, j = 1, \ldots, n_h$ are the associated nodal bases. It is then straightforward to see that $\{ \varphi_{i,j}(x, \hat{y}) := \xi^H_i(x) \cdot \psi^h_j(\hat{y}) \mid i = 1, \ldots, N_H, j = 1, \ldots, n_h \}$ forms a nodal basis of the conforming tensor product FE space

$$V^{H \times h} := \left\{ v^{H \times h} \in C^0(\hat{\Omega}) \mid v^{H \times h}|_{T_{i,j}} \in \mathbb{P}_{k,l} | T_{i,j} \in \hat{T} \right\} \subset V,$$

where $\mathbb{P}_{k,l}$ is defined as $\mathbb{Q}_{k,l} := \{ \sum_j c_j v_j(x) w_j(\hat{y}) \mid v_j \in \mathbb{P}_{k,l}, w_j \in \mathbb{P}_{1,1} \}$. We remark that since $Y^h_m \subset Y^h$, there holds $V^h_m \subset V^{H \times h}$ and we denote by $V^{H \times h}$ the reference FE space. The reference FE approximation of the full problem (2.1) then reads as follows:

$$p^{H \times h} \in V^{H \times h} : \quad a(p^{H \times h}, v^{H \times h}) = f(v^{H \times h}) \quad \forall v^{H \times h} \in V^{H \times h}.$$ 

The model error $e^m := p^{H \times h} - p^m$ satisfies $a(e^m, v^{H \times h}) = r_m(v^{H \times h})$ for all $v^{H \times h} \in V^{H \times h}$, where $r_m(v^{H \times h}) := \int (v^{H \times h} - a(p^m, v^{H \times h})$ for all $v^{H \times h} \in V^{H \times h}$. The corresponding Riesz representor $R_m^{H \times h}$ is given by $(R_m^{H \times h}, v^{H \times h})_V = r_m(v^{H \times h})$ for all $v^{H \times h} \in V^{H \times h}$, where the $V$-inner product has been defined in section 2.

**Proposition 4.1** (a posteriori error bound). The error estimator $\Delta_m$ defined as

$$\Delta_m := \| R_m^{H \times h} \|_V / c_0,$$

satisfies

$$\| p^{H \times h} - p^m \|_V \leq \Delta_m \leq c_1 \| p^{H \times h} - p^m \|_V,$$

where $c_0$ and $c_1$ have been defined in section 2.

**Proof.** We refer to the RB literature for the proof of this standard result (see, e.g., [36]).
5. Analysis of the computational costs of the HMR-RB approach. In this section we compare the total computational costs for the computation of the discrete reduced solution $p^h_{m}$ using the HMR-RB approach for the computation of the associated reference solution $p^{H \times h}$ of (4.2). For the sake of clarity we consider $X^H = \{ v^H \in C^0(\Omega_D) : v^H|_{T_i} \in P_1(T_i), T_i \in T_H \}, Y^h = \{ v^h \in C^0(\Omega) : v^h|_{T_j} \in P_1(T_j), T_j \in \mathcal{T}_h \}$, and $V^{H \times h} = \{ v^{H \times h} \in C^0(\hat{\Omega}) : v^{H \times h}|_{T_{i,j}} \in Q_{1,1}, T_{j} \in \hat{T} \}$. The computation of the reduced solution $p^h_{m}$ can be decomposed in an $n_h$-dependent stage where we construct the reduction space $Y^h_{m}$ and a subsequent $n_h$-independent stage where the coupled system (2.6) is solved. The total computational costs for computing $p^h_{m}$ include the costs for both stages and can be derived as follows.

1. Construction of $Y^h_{m}$ by Algorithm 2 (Adaptive-HMR-RB). First, we assemble the matrices and right-hand sides of the linear systems of equations for the computation of the snapshots $P^h(\mu)$ (3.6) and the reduced solutions $p^H_{m}$ in $\mathcal{O}(n_h)$ and $\mathcal{O}(N_H)$ operations, respectively. Reusing these matrices we assemble the $V$-inner product for the computation of the error estimator $\Delta_m$ in $\mathcal{O}(n_h N_H')$ operations. In Algorithm 2 Adaptive-HMR-RB, we compute the set of snapshots $P^h(\mu), \mu \in \Xi$, in $\mathcal{O}(n_{\text{train}} n_h)$ operations by a Thomas algorithm [35]. The associated reduced solutions $p^H_{m}$ are computed in $\mathcal{O}(n_{\text{train}} N_H')$ operations by a preconditioned conjugate gradient (pcg) or preconditioned stabilized biconjugate gradient (bi-cgstab) method, which scaled linearly in the offline stage for the considered test cases in section 6. We then compute the error estimators $\Delta_m$ in $\mathcal{O}(n_{\text{train}} n_h N_H')$ operations using a pcg method. The costs for the PODs in Algorithm 1 are dominated by the costs for the POD after the refinement of the parameter space which requires $\mathcal{O}(n_{\text{train}}^2 n_h)$ operations for the formation of the correlation matrix and $\mathcal{O}(n_{\text{train}}^2)$ operations for the solution of the eigenvalue problem. Finally, we compute the integrals in $y$-direction in (2.6) depending on the basis functions $\{ \phi_i \}_{i=1}^m$ in $\mathcal{O}(n_h^2)$ operations.

2. Solution of the coupled system (2.6). We assemble the coupled system (2.6) in $\mathcal{O}(m^2 N_H)$ operations. The solution of the resulting linear system of (2.6) with a pcg (section 6, test case 2) or bi-cgstab method (section 6, test case 3) required $\mathcal{O}(N_H m^2)$ and $\mathcal{O}(N_H^3 m^2)$ operations, respectively.

Analyzing these results, we detect a threshold due to the mesh size independent factor $n_{\text{train}}$. If $N_H$ does not depend on $N_H$, the costs for the construction of $Y^h_{m}$, which are dominated by the costs for the error estimator, amount to $\mathcal{O}(n_{\text{train}} n_h)$ operations. We thus expect a linear scaling of the HMR-RB approach in max$\{n_h, N_H\}$ if $N_H$ is chosen constant, which is demonstrated by the numerical experiments in the next section 6.

For the computation of the reference solution $p^{H \times h}$ we assemble the linear system of equations in $\mathcal{O}(N_H n_h)$ operations. Its solution with a pcg (section 6, test case 2) or bi-cgstab method (section 6, test case 3) required $\mathcal{O}(N_H^2 m^2)$ and $\mathcal{O}(N_H^3 m^2)$ operations, respectively. We conclude that if $n_{\text{train}}$ is sufficiently small compared to the mesh sizes $N_H$ and $n_h$, we expect that starting from a certain mesh size the HMR-RB approach outperforms the bilinear FEM. Although the sample size has to increase for more complex problems, we anticipate that in these situations also the grid resolution and hence $N_H$ has to increase due to the higher complexities, and the HMR-RB approach still outperforms the bilinear FEM. The comparison of the total computational costs of the HMR-RB approach and bilinear FEM in test cases 2 and 3 of section 6 supports these two claims. Finally, we discuss the memory aspect. For the bilinear FEM in particular, a sparse matrix with $9n_h N_H$ nonzero elements has to be stored. For the HMR-RB approach, most storage is needed for the full $n_{\text{train}} \times n_h$-matrix containing the snapshots, and for the sparse matrix of the coupled system with $3m^2 N_H$ nonzero.
elements. Therefore, in general, less memory is needed for the HMR-RB approach than for the bilinear FEM.

6. Numerical experiments. In this section we present several numerical test cases to demonstrate the approximation properties and computational efficiency of the proposed method. First, we study the convergence rate of the HMR-RB approach on a numerical example with an analytical solution, where we observe an exponential order of convergence in the model order \( m \). We compare these rates with the results of the HMR approach with sine functions presented in [32]. The solution of the second test case exhibits little spatial regularity both in the dominant and transverse direction of convergence in the model order \( m \). In the third test case we consider an advection-diffusion equation in a long symmetric channel with sinusoidal wavy walls. Due to a strong advective field, the solution exhibits a main stream and dominant spatial features along the \( x \)-direction. All test cases are computed employing linear FE in \( x \) and \( y \)-direction, i.e., \( X^H = \{ v^H \in C^0(\Omega_1, D) : v_H|_{T_i} \in P_1(T_i), T_i \in T_H \}, \)
\( Y^h = \{ v^h \in C^0(\tilde{\omega}) : v^h|_{T_j} \in P_1(T_j), T_j \in T_h \}, \) and \( V^{H \times h} = \{ v^{H \times h} \in C^0(\tilde{\omega}) : v^{H \times h}|_{T_{i,j}} \in Q_{1,1}, T_{i,j} \in T \} \) using equidistant grids in \( x \)- and \( y \)-direction. If not otherwise stated, we have used one quadrature point in (3.6) and the weights
\[
(6.1) \quad \alpha_l := \frac{x^q_l + x^{q-1}_l}{2} - x_0, \quad \alpha_Q := x_1 - \frac{x^Q_Q + x^Q_{Q-1}}{2}, \quad \alpha_l := \frac{x^q_l - x^{q-1}_l}{2},
\]
l = 2, \ldots, Q - 1 in (3.3), which yield a modified rectangle formula. The relative model error in the \( V \)- or \( L^2 \)-norm is defined as \( \| e_m \|^V := \| e_m \|_V / \| p^{H \times h} \|_V \) or \( \| e_m \|_{L^2(\Omega)} := \| e_m \|_{L^2(\Omega)} / \| p^{H \times h} \|_{L^2(\Omega)} \), respectively, for \( e_m = p^{H \times h} - p^H \). We take into account the discretization error, we compare the discrete reduced solution \( p^H \) with the exact solution \( p \) of (2.1) (test case 1) or with a very finely resolved bilinear FE solution \( p_{\text{fine}} \) (test cases 2 and 3). The relative total error \( \| e \|_V := \| e \|_V := \| e \|_{V} := \| p - p^H \|_V / \| p \|_V \) and \( \| e \|_{V} := \| e \|_V := \| e \|_{V} := \| p_{\text{fine}} - p^H \|_V / \| p_{\text{fine}} \|_V \) for the \( L^2 \)-norm, respectively. We also define a relative error estimator \( \Delta_m := \Delta_m / \| p^{H \times h} \|_V \). To discuss the decay behavior of the coefficient functions, we introduce \( e^{M}_{m} := (\sum_{j=m+1}^{M} \| p_j^{H \times h} \|_{L^2(\Omega_1, D)}^2)^{1/2} \) and \( e^{H}_{m} := (\sum_{j=m+1}^{M} \| \partial_x p_j^{H \times h} \|_{L^2(\Omega_1, D)}^2)^{1/2} \), where \( M := \dim(M^0_2) \) (3.7). We will see below that we usually have \( M \ll n_{\text{train}} \).

**Test case 1.** First, we consider a numerical example with an analytical solution. We choose test case 2 of [32] in order to compare the convergence behavior of our new HMR-RB approach with the one of the HMR framework introduced in [16, 32]. We solve a Poisson problem on \( \Omega = (0, 2) \times (0, 1) \). The analytical solution is chosen to be \( p(x, y) = y^2(1 - y)^2(0.75 - y)(2 - x) \exp(2\pi x) \). The error values of the HMR ansatz taken from [31] and have been divided by \( \| p \|_V \) to obtain relative errors. We use the same mesh sizes \( h \) in \( x \)-direction as in [31] and choose \( h = H \).

If we analyze the behavior of the model error \( \| e_m \|^V \) of the HMR-RB approach (Figure 2(a)), we see that \( \| e_m \|^V \) converges exponentially fast in \( m \). In contrast, the usage of sine functions in the orthonormal expansion of the HMR approach excludes a priori exponential convergence rates also for smooth functions like \( C^\infty \)-functions [9]. The expected convergence rate of the model error for the present example is \( m^{-1} \) [10]. This rate can be detected for a sufficiently small \( H \) (here \( H \leq 0.025 \)) in Figure 2(b), where the total error \( \| e \|^V \) of the HMR ansatz is depicted, as for \( H \leq 0.025 \) the model error dominates the discretization error. Hence, the slope of \( \| e \|^V \) most closely approximates the one of \( \| e_m \|^V \) for \( H = 0.0125 \). Regarding the convergence behavior
of the total error \( \|e\|_V^{rel} \) of the HMR-RB approach (Figure 2(c)), we see that for \( m \geq 3 \) or even \( m \geq 2 \) the discretization error dominates the model error. For the HMR ansatz and \( H \leq 0.05 \), this is not even the case for a model order \( m = 16 \).

**Test case 2.** In this test case we demonstrate that the HMR-RB approach may potentially approximate a nonsmooth, full solution exponentially fast. We consider \( \Omega = (0,1.1) \times (0,1.1) \) and a diffusion tensor

\[
K = \begin{pmatrix}
0.25 & 0 \\
0 & 25
\end{pmatrix}.
\]

The source term \( s \) is depicted in Figure 3 and defined in (A.1) in the appendix. We emphasize that \( s \in H^1(\Omega) \cap C^0(\Omega) \) but \( s \notin H^2(\Omega) \). On \( \Gamma_0 \) —defined in (2.2)—we prescribe homogeneous Neumann boundary conditions and on the remaining part of \( \partial \Omega \) homogeneous Dirichlet boundary conditions. The reference solution \( p^{H \times h} \) for \( N_H = n_h = 800 \) is displayed in the first picture of Figure 4 and shows a stronger variation in \( x \)-direction. We have done a convergence study to ensure that \( p^{H \times h} \) contains all essential features of the exact solution. Comparing with the corresponding HMR-RB approximation \( p^m_H \) for \( m = 3, 5, 7, N_H = n_h = 800 \), and \( N_H = 80 \) (Figure 4), we see a very good visual agreement of \( p^{H \times h} \) and \( p^H \).

Next we analyze the approximation properties of the HMR-RB approach from a quantitative viewpoint. First, we observe in Figure 5(a) that \( \|e_m\|_V^{rel} \) converges exponentially fast in \( m \) with the same rate as \( \varepsilon_m^{H^1} \). We remark that by exploiting the fact that \( V^{H \times h} \) is a finite dimensional space, it is always possible to derive an exponential rate, which, however, depends on the mesh size of the discretization (cf. [8] for the proof of this statement for the PGD approach). As the convergence rate of \( \|e_m\|_V^{rel} \) in Figure 5(a) does not change for decreasing mesh size and stays the same even for very fine meshes, we argue that this exponential convergence rate does not result from the fact that \( V^{H \times h} \) is of finite dimension. Moreover, also \( \|e_m\|_{L^2(\Omega)} \) (3.11), and \( \varepsilon_m^{L^2} \) exhibit the same exponential convergence rate (Figure 5(b)). The same holds true for the eigenvalues of the POD \( \lambda_m \) and \( \|p^H_m\|_{L^2(\Omega \cup D)}^2 \) (Figure 5(c)). Therefore we infer that the convergence behavior of the POD transfers to the coefficients \( \|p^H_m\|_{L^2(\Omega \cup D)}, \varepsilon_m^{L^2}, \varepsilon_m^{H^1} \), and to the model error \( \|e_m\|_V^{rel} \). We may hence conclude that the discrete solution manifold \( \mathcal{M}_m^H \) (3.7) and the reference solution \( p^{H \times h} \) are approximated with the same approximation quality by the reduction space \( Y_m^H \) for the present
Fig. 3. Test case 2: Source term $s(x,y)$ defined in (A.1), appendix.

Fig. 4. Test case 2: In comparison from left to right and top to bottom: the reference 2D bilinear FE solution $p^{H \times h}$ and the discrete reduced solution $p^{Hm}$ using 3, 5 and 7 basis functions.

Fig. 5. Test case 2: Illustration of the model error convergence (a) $\|e_m\|_{V^1}^{rel}$ and $\|\bar{e}_m\|_{V^1}^{H}$ for $H = 0.001375$, (b) $\|e_m\|_{L^2(\Omega)}^{rel}$, $\|\bar{e}_m\|_{L^2(\Omega)}^{POD}$ and $\|\bar{e}_m\|_{L^2(\Omega)}^{L^2}$ for $H = 0.001375$, and (c) $\lambda_m$ and $\|p_m\|_{L^2(\Omega)}^{H}$ for $H = 0.001375$.

test case. Analyzing the convergence behavior of the total error $\|e\|_{V}^{rel}$ in Figure 6, we detect an interaction of the model error and the discretization error. Up to a certain model order $m$, for example, $m = 8$ for $H = 0.001375$, the model error clearly dominates the discretization error. Then the proportion of the discretization error increases and finally dominates the model error for higher orders of $m$ (for instance, $m \geq 11$ for $H = 0.001375$). We also observe in Figure 6 that the total error $\|e\|_{V}^{rel}$ converges linearly in $H$, which is the expected rate.

To demonstrate the computational efficacy of the HMR-RB approach, we compare the total computational costs of the latter with the costs for the computation of the reference solution $p^{H \times h}$ of (4.2). We emphasize that the costs of the HMR-RB
approach depicted in Figure 7 include the costs for the construction of the reduction space $Y_m^H$. To enable a fair comparison, we have used the same routines for the assembling of matrixes and right-hand sides and a pcg method with the same settings for the solution of the linear systems of equations. The tolerance for the POD (see section 3.2) has been chosen as $\varepsilon_{tol} = 10^{-5}$. In the left picture of Figure 7, we observe that the computational costs for the bilinear FEM scale at least quadratically in $N_H (= n_h)$. In contrast, the HMR-RB approach scales linearly in $N_H$ for $N_H = 10$ and a bit worse than linearly for $N_H = N_H/10$. This confirms the theoretical run-time complexities derived in section 5. Also the supposed threshold due to the factor $n_{train}$ can be detected. The average run-times for $N_H = N_H/10$ lie well above the ones for $N_H = 10$ for $N_H = 400, 800$ due to the higher costs of Algorithm 1 \textsc{AdaptiveTrainExtension} for higher values of $N_H$. As $N_H = 10$ yields the same relative total error in a much shorter run-time than $N_H = N_H/10$ (Figure 7), we conclude that it is sufficient and most efficient to use only $N_H = 10$ finite elements in the dominant direction in Algorithm 1 for the present example. Finally, we emphasize that for $\|e_m\|_V^{rel} \leq 10^{-2}$, the HMR-RB approach with $N_H = 10$ clearly outperforms the bilinear FEM for $N_H = n_h$ and also when using a coarser discretization in $y$-direction with $n_h = 0.5N_H$ in the bilinear FEM (Figure 7). For smaller values of $\|e_m\|_V^{rel}$, the gain of using the HMR-RB approach instead of the standard bilinear FEM increases.

All results that have been reported for this test case so far have been computed for the following choice of the input parameters of Algorithm 2 \textsc{Adaptive-HMR-RB}: $G_0 = [0, 0.275, 0.55, 0.825, 1.1] \times [-0.1, 0.1] \times [-1, 1], m_{max} = 2, i_{max} = 1, n_\Xi = 6$, and $\theta = 0.05$. Based on the obtained numerical results, we choose $\sigma_{thres} = (i_{max} - 1) \cdot [\text{diag}(g)] + 1$ to achieve $(i_{max} - 1) \cdot \text{diag}(g) \leq \sigma_{thres} \leq i_{max} \cdot [\text{diag}(g)]$, where $\text{diag}(g)$ is the diagonal of an element of the initial grid $G_0$, defined in section 3.2, and $\lfloor \cdot \rfloor$ denotes the ceiling function. By comparing $\|e_m\|_V^{rel}$ for different values of $I_0$ and $I_1$—defined in section 3.1—we have observed that the choice of the intervals $I_0$ and $I_1$ has hardly any effect on the outcome of the method. Regarding the choice of $m_{max}$, $i_{max}$, and $n_\Xi$, we observe a low sensitivity of $\|e_m\|_V^{rel}$ with respect to a change in the input parameters for a fixed sample size of approximately 220, where $n_{train} \approx 280$ for $m_{max} = 4$, $i_{max} = 1$, and $n_\Xi = 1$ (Figure 8(a)). Nevertheless, we detect that the convergence rate becomes slightly better from $m_{max} = 1$ to $m_{max} = 2$, etc.
but a further increase produces no additional improvement. Regarding $i_{\text{max}}$, it can be stated that an increase of $i_{\text{max}}$ deteriorates the convergence rate. The associated total computational costs for $N_H = n_h = 200$ in Figure 8(b) support these findings as the choice of $i_{\text{max}} = 1$ always performs best by yielding a better relative error for comparable run-times. As the choice $m_{\text{max}} = 3$ performs worse than $m_{\text{max}} = 2$ and the choice $m_{\text{max}} = 4$ reduces the relative error only at considerably additional cost, we conclude that $m_{\text{max}}$ should be chosen either equal to one or two and that $i_{\text{max}} = 1$ is the best choice. Fixing $m_{\text{max}} = 2$ and $i_{\text{max}} = 1$, we see in Figure 8(c) that the convergence rate of $\|e_m\|^{rel}_V$ gets worse for growing $n_{\Xi}$, whereas $M = \text{dim}(\mathcal{M}_{\Xi})^n$ increases. The same behavior can be observed for the other combinations of $m_{\text{max}}$ and $i_{\text{max}}$ listed in Figure 8(a), except for $m_{\text{max}} = 1$ and $i_{\text{max}} = 1$, where the model error improves for increasing $n_{\Xi}$ until $n_{\Xi} = 6$. We further see in Figure 8(c) that for $n_{\Xi} \leq 3$ a tolerance of $10^{-3}$ can in general not be achieved, as the number $M$ of linear independent snapshots in $\mathcal{M}_{\Xi}$ is too small. Hence, for $m_{\text{max}} = 2$ and $i_{\text{max}} = 1$, we choose $n_{\Xi} \geq 5$ to ensure that the snapshot set $\mathcal{M}_{\Xi}$ is rich enough.

**Test case 3.** This test case, which is very similar to test case 4 in [32], comes from the field of hemodynamics, modeling a Bellhouse membrane oxygenator for extracorporeal circulation (cf. [4]). We model oxygen transport within a symmetric channel with sinusoidal wavy walls, which is a typical geometry in this context. To this end we define $\Omega$ as the domain which is bounded by the functions $x = 0$, $y = 1 - 0.25 \sin(2\pi x)$, $x = 4$, and $y = 2 + 0.25 \sin(2\pi x)$ and consider

\begin{equation}
\frac{\partial p}{\partial t} = \nabla p \cdot \nabla v + \int_{\Omega} (100, 0)^t \cdot \nabla pv \quad \text{and} \quad f(v) = 0,
\end{equation}

where $p$ models the oxygen concentration in the blood. We prescribe nonhomogeneous Dirichlet boundary conditions on the inflow boundary $\Gamma_0$ by setting $p(0, y) = \sqrt{2} \sin(\pi(y + 1))$, homogeneous Neumann boundary conditions on the outflow boundary $\Gamma_1$, and homogeneous Dirichlet boundary conditions on $\Gamma_*$.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure8}
\caption{Test case 2: $\|e_m\|^{rel}_V$ for $D = [0, 1.1] \times [-1, 0] \times [-1, 1]$ for different combinations of inputs of Algorithm 2 Adaptive-HMR-RB: Varying $[m_{\text{max}}, i_{\text{max}}, n_{\Xi}]$ for $n_{\text{train}} \approx 220$ (a) with associated total run-times (b); $m_{\text{max}} = 2, i_{\text{max}} = 1$, and varying $n_{\Xi}$ (c).}
\end{figure}
Fig. 9. Test case 3: In comparison from left to right and top to bottom: the reference 2D bilinear FE solution $p^{H \times h}$ and the discrete reduced solution $p^H_{M}$ using 6, 9, and 12 basis functions; $N_H = 1200, n_h = 400, N_{H'} = 20$.

Fig. 10. Test case 3: Adaptively refined training set after the application of Algorithm 2 Adaptive-HMR-RB.

$H = 1/3 \cdot 10^{-3}$ yields a local Péclet number which is for the prescribed advection field strictly less than 1. For local Péclet numbers greater than 1, no oscillations have been observed either, indicating that also for these discretizations the scheme is stable. Therefore no stabilization scheme has been used. A comparison with the corresponding HMR-RB approximation shows that $p^H_6$ reproduces the behavior of $p^{H \times h}$ for $0 \leq x \leq 0.75$ quite well but shows a bad approximation quality for $x \geq 2$ (Figure 9). This is due to a major refinement of $\Xi$ in the interval $[0.5, 1]$ during Algorithm 2 Adaptive-HMR-RB (Figure 10). $p^H_9$ already captures the behavior of the main stream very well but exhibits some oscillations caused by the irregular shape of $\Omega$. We finally see a very good visual agreement of $p^{H \times h}$ and $p^H_{12}$. Figure 10 shows that during Algorithm 2, the parameter space $D = [0, 4] \times [-2, 2] \times [-5, 5]$ is mostly refined at the narrowing of $\Omega$ at $x = 0.75$. The other parts of $D$ are refined only once due to the second indicator $\sigma(g)$. We therefore conclude that $\eta(g)$ causes a refinement of the relevant parts of $D$ and that Algorithm 2 Adaptive-HMR-RB is able to detect recurring structures in the full solution $p$. 

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.
Fig. 11. Test case 3: Comparison of the model error convergence for different quadrature rules for $N_{H'} = N_{H}/10$ in the $V$-norm/$H^1$-semi-norm: (a) $\| e_m \|_V^{rel}$ for different $H$ and $e_m^{H'_1}$ for $H = 0.00333$ for the one-point rectangle formula (1qp), and for $H = 0.00333$ in the $L^2$-norm: (b) $\| e_m \|_L^2(\Omega)$ and $e_m^{POD}$ for the one-point- and two-point-rectangle formula (2qp), (c) $\| e_m \|_V^{rel}$ for the one-point-, two-point-rectangle formula and the trapezoidal rule, (d) $\lambda_m$ and $\| \bar{p}_m \|_{L^2(\Omega_{1D})}^{2}$ for one-point-, two-point-rectangle formula and the trapezoidal rule (trapez), (e) relative total error $\| e \|_V^{rel}$ for different $H$ and $\| e_m \|_V^{rel}$ for $H = 0.00333$ and (f) $\Delta_m^{rel}$ for different $H$ and $\| e_m \|_V^{rel}$ for $H = 0.00333$ both for the one-point-rectangle formula.

Studying the convergence behavior of $\| e_m \|_V^{rel}$ (Figures 11(a), 11(c)) and $\| e_m \|_{L^2(\Omega)}^{rel}$ (Figure 11(b)) for different quadrature formulas (3.3), we detect an error plateau which gets significantly smaller when we increase the number of quadrature points in...
(3.3). After the plateau $\|e_m^e\|^2_{V_L^t}$ and $\|e_m^e\|^2_{L^2(\Omega)}$ converge exponentially fast and the convergence rates of $\|e_m^e\|^2_{L^2(\Omega)}$ and $e_m^e$ coincide (Figure 11(b)) for all considered quadrature rules. Finally the $L^2$-norm of the coefficients $\|p_m^H\|^2_{L^2(\Omega;D)}$ (Figure 11(d)) increases until $m = 5$ for one quadrature point in (3.3) and hence their convergence behavior significantly differs from $\lambda_m$ for smaller $m$. For two quadrature points, we detect only a slight increase and we observe for all considered quadrature rules that the convergence rates of $\|e_m^H\|^2_{L^2(\Omega)}$, and $\lambda_m$ coincide after the plateau. Altogether we conclude that due to the strong advective field $b$, which is reinforced by the transformation $\Psi$ (Figure 1), the discrete solution manifold $M_m^b$ (3.7) does not contain all essential features of the solution (in contrast to test case 2) for one quadrature point. This might also explain the flattening of $\|e_m^H\|^2_{L^2(\Omega)}$ for $m = 14$ in Figure 11(d) and $e_m^H$ in Figure 11(a), which is due to the fact that $a(\cdot, \cdot)$ (6.3) is nonsymmetric. For higher values of $m$ ($m \geq 8$) the effectiveness indices are very close to 1. Next we compare the total computational costs for the computation of the reference solution $p^{H \times h}$ of (4.2) and the reduced solution $p_m^H$, where the costs for the latter also include the costs for the construction of the reduction space $Y_m$. The tolerance for the POD (see section 3.2) has been set to $\varepsilon_{tol} = 10^{-4}$. Owing to the geometry of $\Omega$, we have chosen $n_h = 1/3 N_H$. We observe in Figure 12, left picture, that the computation of $p^{H \times h}$ requires between $O(N^2_H)$ and $O(N^3_H)$ operations, whereas the HMR-RB approach scales linearly in $N_H$. Also, for this test case, the theoretical computational costs derived in section 5 are confirmed and a threshold due to the factor $n_{train}$ can be detected. Moreover, we see that the costs for the one-point-rectangle formula and the three-point-trapezoidal rule are about the same and that they are significantly higher than the costs for the two-point-rectangle formula (Figure 12). As all three quadrature rules yield the same relative total error $\|e\|^2_{V_L^t}$, we conclude that the two-point-rectangle formula performs best for the present test case. Since $N_H = 10$ yields the same relative total error at lower costs than $N_H = 20$ for all three quadrature rules, we further infer that it is sufficient and most efficient to choose also for this numerical example $N_H = 10$. Finally, for $\|e\|^2_{V_L^t} \leq 0.03$, the HMR-RB approach with $N_H = 10$ clearly outperforms the bilinear FEM for $n_h = 1/3 N_H$ and also for the coarser discretization in $y$-direction with $n_h = 1/4 N_H$ for the bilinear FEM. Moreover, the advantage of using the HMR-RB approach instead of the standard bilinear FEM increases significantly for decreasing $\|e\|^2_{V_L^t}$. We emphasize that although $\Xi$ is twice as large as in test case 2, the HMR-RB approach outperforms the bilinear FEM even for higher values of the relative total error $\|e\|^2_{V_L^t} = 0.03$ in the present example and 0.01 in test case 2—as the boundary layers (Figure 9) require in relation to test case 2 a much higher grid resolution. As Algorithm 1 ADAPTIETRAINEXTENSION is able to detect recurring structures in the full solution (Figure 10) and thus possibly limits the growth of the sample size also for more complex problems to some extent, we expect that the increase of $n_{train}$. and
the grid resolution balance for more complex problems as in the present test case, and hence the HMR-RB approach still outperforms the bilinear FEM.

Finally, we address the choice of the input parameters of Algorithm 2 Adaptive-HMR-RB. First, we detect a stronger sensitivity of $\|e_m\|_V^{rel}$ with respect to the intervals $I_0$ and $I_1$ (Figure 13(b)) than in test case 2. However, except for $D = [0, 4] \times [−2, 2] \times [−5, 5]$, all choices result in the same convergence rate after the error plateau and yield about the same gain when increasing the accuracy of the quadrature formula. Furthermore, we have observed that all considered parameter domains yield the same total error $\|e\|_V^{ref}$ at less or equal computational costs than $D = [0, 4] \times [−2, 2] \times [−5, 5]$. The differences in the required computational costs between the different choices of $D$ are relatively small—for instance, one second between the smallest and the highest run-time for $N_H = 1200$, $n_h = 400$, and $N_{H'} = 20$. We hence conclude that in spite of the relatively high sensitivity of the convergence rate $\|e_m\|_V^{rel}$ with respect to the values of $I_0$ and $I_1$, the choice of the latter has relatively little influence on the performance of the HMR-RB approach in terms of computa-

![Figure 12](image_url)

**Fig. 12.** Test case 3: Comparison of the run-time [s] of bilinear FEM and the HMR-RB approach ($N_{H'} = 10, 20$) for the one-point- ($1qp$) and two-point-rectangle formula ($2qp$) and the trapezoidal rule (trap.) for $n_h = 1/3N_H$ (left) and per $\|e\|_V^{rel}$ (middle and right).

![Figure 13](image_url)

**Fig. 13.** Test case 3: $\|e_m\|_V^{rel}$ for different parameter spaces $D = \alpha D \times \cdots$ (a); $\|e_m\|_V^{rel}$ for $D = [0, 4] \times [−2, 2] \times [−5, 5]$ for different combinations of inputs of Algorithm 2 Adaptive-HMR-RB: varying $[m_{\text{max}}, i_{\text{max}}, n_{\Xi}]$ for $n_{\text{train}} \approx 400$ (b).
of the parametrized 1D problems is able to transfer the essential transverse features as respects to a change in the input parameters for a fixed sample size of approximately 400 (Figure 13(b)). Nevertheless, we observe an improvement of the convergence rate for increasing \( m_{\text{max}} \) (Figure 13(b), left picture) and except for \( m_{\text{max}}=1, i_{\text{max}}=1 \) also an improvement for growing \( i_{\text{max}} \) (Figure 13(b), right picture). The corresponding computational costs for fixed total error \( ||e||_{V,I}^e \) for growing \( m_{\text{max}} \) increase, whereas for \( i_{\text{max}} \) no trend is detectable. As the strong variation for \( m_{\text{max}}=1 \) makes it difficult to determine a suitable choice of \( n_\Xi \), \( m_{\text{max}}=2 \) seems to be preferable. For all combinations of \( m_{\text{max}} \) and \( i_{\text{max}} \) listed in Figure 13(b), we have observed a deterioration of the convergence rate of \( ||e_m||_{V,I}^e \) and a concurrent increase of \( M \) for growing \( n_\Xi \). All computations for this test case have been conducted for \( m_{\text{max}}=2, i_{\text{max}}=3 \), and \( m_{\text{max}}=5 \) for the one-point-rectangle formula, where choosing \( i_{\text{max}} \geq 4 \) ensures that \( \dim(Y_m^h) \) is big enough. For the two-point-rectangle formula and for the trapezoidal rule we have chosen \( m_{\text{max}}=i_{\text{max}}=1, n_\Xi=5 \) and \( m_{\text{max}}=2, i_{\text{max}}=1, n_\Xi=5, 6 \), respectively, and \( \theta=0.025 \) and \( \sigma_{\text{thres}}=i_{\text{max}}[\text{diag}(g)]+1 \).

We conclude the numerical experiments with some remarks on the choice of the input parameters of Algorithm 2. Test cases 1 and 2 demonstrate that it seems to be sufficient to use the one-point-rectangle formula (6.1) for purely diffusive problems. In contrast for advection-dominated problems, as in test case 3, quadrature rules of higher accuracy should be used, where the two-point-rectangle formula seems to be preferable to the three-point-trapezoidal rule. Using the one-point-rectangle formula \( m_{\text{max}}=2, i_{\text{max}}=2 \), and \( n_\Xi=5, \ldots, 8 \) yielded a good result for all considered test cases. For the two-point-rectangle formula \( m_{\text{max}}=1, i_{\text{max}}=1, n_\Xi=4, \ldots, 8 \) performed well. We thus conclude that these values might be suitable a priori choices for the input parameters of Algorithm 2 for the respective settings.

7. Conclusions. We have introduced a dimensional reduction approach which uses a highly nonlinear approximation based on RB methods to determine the reduction space \( Y_m \) in the HMR framework. Starting from the full problem, initially a parametrized lower dimensional problem in the transverse direction has been derived. Here, the parametrization enabled to include the unknown behavior of the full solution in the dominant direction in \( Y_m \). The generation of snapshots is based on a reliable and efficient a posteriori error estimator combined with an adaptive training set extension. Finally, a POD is applied to the set of snapshots to select the principal components, which form the reduction space \( Y_m \). The numerical experiments demonstrate that the set of solution snapshots and the reference solution are approximated by the reduction space with the same accuracy for the purely diffusive problems. For problems with a dominant advective term a comparable approximation rate can be achieved by using quadrature rules of higher accuracy when deriving the parametrized problem. Thus, we conclude that the proposed ansatz for the derivation of the parametrized 1D problems is able to transfer the essential transverse features of the solution to the solution manifold. Furthermore, numerical experiments demonstrate that the new approach converges exponentially fast with respect to the model order \( m \) for problems with smooth solutions as well as for a test case where the source term is only continuous. Our new approach shows faster convergence, with respect to the model order, than the classical HMR ansatz using trigonometric polynomials in all test cases. An analysis of the computational costs shows that the HMR-RB approach scales linearly in the number of degrees of freedom used for the computations in the
dominant direction, whereas the corresponding FE reference approximation exhibits at best a quadratic scaling. Finally, we demonstrated that the new approach is indeed particularly beneficial for problems that exhibit a main stream and hence a dominant spatial direction.

Appendix. Definition of the source term \( s \) of test case 2. The source term \( s \) used in test case 2 and depicted in Figure 3 is defined as follows:

\[
s(x, y) := 1.5 e^{-5x} \cdot \sin(\pi y) + \sum_{i=1}^{5} s_{x,i}(x) \cdot s_{y,i}(y)
\]

with

\[
s_{x,1}(x) := \begin{cases} 
-800x^2 + 120x - 2.5, & 0.025 \leq x \leq 0.125, \\
-800x^2 + 1000x - 310.5, & 0.125 \leq x \leq 0.225, \\
0, & \text{else}
\end{cases}
\]

\[
s_{y,1}(y) := -100y^2 + 30y - 1.25, 
\quad \text{for } 0.05 \leq y \leq 0.25,
\]

\[
s_{x,2}(x) := \begin{cases} 
-400x^2 + 140x - 11.25, & 0.125 \leq x \leq 0.225, \\
-800x^2 + 1160x - 418.5, & 0.225 \leq x \leq 0.775, \\
0, & \text{else}
\end{cases}
\]

\[
s_{y,2}(y) := -200y^2 + 280y - 96, 
\quad \text{for } 0.6 \leq y \leq 0.8,
\]

\[
s_{x,3}(x) := \begin{cases} 
-560x^2 + 308x - 40.95, & 0.225 \leq x \leq 0.325, \\
-800x^2 + 1320x - 542.5, & 0.325 \leq x \leq 0.875, \\
0, & \text{else}
\end{cases}
\]

\[
s_{y,3}(y) := -140y^2 + 70y - 7.35, 
\quad \text{for } 0.15 \leq y \leq 0.35,
\]

\[
s_{x,4}(x) := \begin{cases} 
-400x^2 + 300x - 55.25, & 0.325 \leq x \leq 0.425, \\
-800x^2 + 1480x - 682.5, & 0.425 \leq x \leq 0.975, \\
0, & \text{else}
\end{cases}
\]

\[
s_{y,4}(y) := -200y^2 + 180y - 38.5, 
\quad \text{for } 0.35 \leq y \leq 0.55,
\]

\[
s_{x,5}(x) := \begin{cases} 
-800x^2 + 760x - 178.5, & 0.425 \leq x \leq 0.525, \\
-800x^2 + 1640x - 838.5, & 0.525 \leq x \leq 1.075, \\
0, & \text{else}
\end{cases}
\]

\[
s_{y,5}(y) := -100y^2 + 190y - 89.25, 
\quad \text{for } 0.85 \leq y \leq 1.05.
\]

Acknowledgment. We thank Prof. S. Perotto for fruitful discussions.

REFERENCES


