Nonlinear Methods for Model Reduction

Andrea Bonito, Albert Cohen, Ronald DeVore,
Diane Guignard, Peter Jantsch, and Guergana Petrova [*]

May 6, 2020

Abstract

The usual approach to model reduction for parametric partial differential equations (PDEs) is to construct a linear space V_n which approximates well the solution manifold $\mathcal{M}$ consisting of all solutions $u(y)$ with y the vector of parameters. This linear reduced model V_n is then used for various tasks such as building an online forward solver for the PDE, estimating the state or parameters from data observations. It is well understood in other problems of numerical computation that nonlinear methods such as adaptive approximation, n-term approximation, and certain tree-based methods may provide much improved numerical efficiency. This suggests the use of nonlinear methods for model reduction as well. A nonlinear method would replace the linear space V_n by a nonlinear space Σ_n. This idea has already been suggested in recent papers on model reduction [11, 13, 16] where the parameter domain is decomposed into a finite number of cells and a linear space of low dimension is assigned to each cell.

Up to this point, results on such a nonlinear strategy are ad hoc and there is little known in terms of precise performance guarantees. Moreover, most numerical experiments for nonlinear model reduction have only been performed when the parameter dimension is very small (one or two). In the present paper, a step is made towards providing a more cohesive theory for nonlinear model reduction of the above type. Framing these methods in the general setting of library approximation allows us to give a first comparison of their performance with those of standard linear approximation for any general compact set. We then turn to the study of the application of these methods for solution manifolds of parametrized elliptic PDEs. In this context, we study a very specific example of library approximation where the parameter domain is split into a finite number N of rectangular cells and where different reduced affine spaces of dimension m are assigned to each cell. The performance of this nonlinear procedure is analyzed from the viewpoint of accuracy of approximation versus m and N. A concrete strategy for the subdivision of the parameter domain is given and it is shown how this subdivision can be exploited for various numerical tasks.

1 Introduction

Complex systems are frequently described by parametrized partial differential equations (PDEs) that take the general form

$$\mathcal{P}(u, y) = 0, \tag{1.1}$$

[*] This research was supported by the NSF grants DMS 1817691 (AB), DMS 1817603 (RD-GP), ONR grant N00014-17-1-2908 (RD); DG was supported by the Swiss National Science Foundation grant P2ELP2-175056 and IAMCS at TAMU, and PJ was supported by an NSF Fellowship DMS 17-04121. A portion of this research was completed while RD (Simon Fellow), DG, and PJ were supported as visitors of the Isaac Newton Institute at Cambridge University

where y ranges over some parameter domain Y, and $u = u(y)$ is the corresponding solution which is assumed to be uniquely defined in some Hilbert space V for every $y \in Y$. We denote by $\|\cdot\| = \|\cdot\|_V$ and $\langle\cdot,\cdot\rangle$ the norm and inner product of V, respectively. In what follows, we assume that the parameters are countably infinite and have been rescaled so that $Y = [-1,1]^{\mathbb{N}}$. The case of a finite dimensional parameter $y = (y_1, \dots, y_J)$ can always be recast in this setting by considering that $u(y)$ does not depend of the variable y_j for $j > J$.

There are three main problem areas associated with parametric PDEs:

(i) building forward solvers to efficiently compute approximations to $u(y)$ for any given $y \in Y$;

(ii) estimating the state $u(y)$ from data observation when the parameter y is unknown;

(iii) estimating the parameter y that can give rise to an observed state u.

One commonly used approach to tackle these three ranges of problems in a numerically efficient way is *reduced modeling*. In its most usual form, it is based on introducing a linear space V_n of low dimension n which is tailored to provide an accurate approximation to all solutions $u(y)$ as y varies in Y, or equivalently, to the *solution manifold*,

$$\mathcal{M} := \{u(y) : \ y \in Y\}. \tag{1.2}$$

1.1 Linear reduced models

There are two common approaches to finding a reduced model V_n. The first one is to establish that the forward map $y \mapsto u(y)$ has a certain analyticity in y, and thereby admits a Taylor series representation

$$u(y) = \sum_{\nu \in \mathcal{F}} t_\nu y^\nu, \quad t_\nu \in V. \tag{1.3}$$

Here $\mathcal{F}$ denotes the set of $\nu = (\nu_1, \nu_2, \dots)$ which have finite support and whose entries are nonnegative integers. Quantitative bounds for the size of the Taylor coefficients t_ν allow one to prove that for each ε, there is a finite set $\Lambda = \Lambda(\varepsilon) \subset \mathcal{F}$ such that

$$\sup_{y \in Y} \|u(y) - \sum_{\nu \in \Lambda} t_\nu y^\nu\|_V \leq \varepsilon. \tag{1.4}$$

The space $V_n := \mathrm{span}\{t_\nu : \nu \in \Lambda\}$ provides the reduced model with $n = \#(\Lambda)$. In this case, an approximation of $u(y)$ in V_n is readily provided by the function

$$\hat{u}(y) := \sum_{\nu \in \Lambda} t_\nu y^\nu, \tag{1.5}$$

that is, using the y^ν as the coefficients of $\hat{u}$ in the basis t_ν. Quantitative bounds on the cardinality of $\Lambda(\varepsilon)$ are known under various assumptions on the coefficients of the PDE [7].

The second approach to finding a reduced model is to judiciously select certain *snapshots* $u(y^1), \dots, u(y^n)$ of u via a greedy procedure, and use the space $V_n := \mathrm{span}\{u(y^1), \dots, u(y^n)\}$ as the reduced model. In this case, the approximation of $u(y)$ in V_n requires a projection step.

Recent results show that there is a numerical advantage in the Taylor coefficient approach to finding a reduced basis, at least in the case of elliptic and certain parabolic PDEs, in the sense that it is sometimes possible to a priori find the set Λ by exploiting the parametric form of the diffusion coefficients [1]. This avoids computationally expensive search algorithms that are a component of greedy reduced basis selections. On the other hand, greedy procedures have the advantage that they are provably near-optimal for finding a linear space to approximate u, in the sense that their convergence rates are similar to those of the optimal linear spaces for approximating $\mathcal{M}$, see [2]. Moreover, as we illustrate further in this paper, numerical experiments show that for a prescribed target accuracy, the greedy generated spaces that meet this accuracy are of significantly lower dimension then their polynomial counterparts.

There is a rigorous theory that quantifies the approximation performance of both of these reduced models; see [7] for a summary of known results. The theory is most fully developed in the case of elliptic PDEs of the form

$$-\operatorname{div}(a\nabla u) = f, \tag{1.6}$$

set on a physical domain $D \subset \mathbb{R}^d$, say with Dirichlet boundary conditions $u_{|\partial D} = 0$, and where the diffusion function a has an affine parametrization

$$a(y) = \bar{a} + \sum_{j\geq 1} y_j \psi_j, \tag{1.7}$$

for some given functions $\bar{a}$ and $(\psi_j)_{j\geq 1}$ in $L^\infty(D)$. These functions are assumed to satisfy the condition

$$\left\| \frac{\sum_{j\geq 1} |\psi_j|}{\bar{a}} \right\|_{L^\infty(D)} < 1, \tag{1.8}$$

which is equivalent to the following assumption.

Uniform Ellipticity Assumption (UEA): There exist $0 < a_{\min} \leq a_{\max} < \infty$ such that

$$0 < a_{\min} \leq a(y) \leq a_{\max} < \infty, \quad y \in Y. \tag{1.9}$$

Lax-Milgram theory then ensures that whenever $f \in V' = H^{-1}(D)$, for each $y \in Y$, the corresponding solution $u(y)$ is uniquely defined in the Hilbert space $V := H_0^1(D)$ endowed with the norm $\|\cdot\|_V := \|\nabla \cdot\|_{L^2(D)}$.

1.2 Nonlinear reduced models

It is known that in many contexts, numerical methods based on nonlinear approximation outperform linear methods, in the sense of requiring a much reduced computational cost to achieve a prescribed error tolerance [9]. This motivates us to consider replacing the linear space V_n by a nonlinear space Σ_n depending on n parameters. We call such a space Σ_n a *nonlinear reduced model.* This idea has already been suggested and studied in certain settings; see e.g. [11, 13, 16]. However, up till now, there has not been a unified study of nonlinear model reduction. The purpose of the present paper is to provide a formal theory for such methods of nonlinear model reduction and to prove some first results that quantify the performance of these nonlinear methods.

The nonlinear reduced models studied in this paper can be placed into the form of what is sometimes called *library approximation.* Given a Banach space X, a library $\mathcal{L}$ is a finite collection of affine spaces $L_1 := x_1 + X_1, \dots, L_N := x_N + X_N$, where each X_j is a linear space of dimension at most m, and each $x_j \in X$, $j = 1, \dots, N$. We set each $X_j = \{0\}$ in the case $m = 0$. For an element $x \in X$, the error of approximation of x by the library $\mathcal{L}$ is

$$E(x, \mathcal{L}) := \inf_{L \in \mathcal{L}} \operatorname{dist}(x, L)_X. \tag{1.10}$$

In other words, given x, we choose the best of the affine spaces $L_j = x_j + X_j$, $j = 1, \dots, N$, to approximate x. Given a library $\mathcal{L}$ and a compact set $K \subset X$, we define

$$E_{\mathcal{L}}(K) := \sup_{x \in K} E(x, \mathcal{L}). \tag{1.11}$$

Here, in the context of reduced models for parametric PDEs, the idea is to keep m small when compared to the dimension n used in linear models V_n, while retaining the same accuracy of the reduced model.

For parametric PDEs, we take $X = V$ and $K = \mathcal{M} := \{u(y) : \ y \in Y\}$ to be the solution manifold of the PDE. A library $\mathcal{L}$ would then consist of affine spaces

$$L_j := u_j + V_j, \tag{1.12}$$

where each $u_j \in V$ and each $V_j \subset V$ has dimension at most m. Then, the best approximation to $u(y)$ from L_j is

$$u_j + P_{V_j}(u(y) - u_j), \tag{1.13}$$

where P_{V_j} is the V-orthogonal projection onto V_j. In this context, when presented with a parameter y for which we wish to compute an online approximation to $u(y)$, the choice of which space L_j to use from a given library $\mathcal{L}$ could be decided in several ways, among which we mention:

(i) searching for a computable bound for $\operatorname{dist}(u(y), L_j)_V = \|u - u_j - P_{V_j}(u(y) - u_j)\|_V$, and choosing the value of j that minimizes this surrogate quantity;

(ii) building an a priori partition of the parameter domain Y into cells Q_j and construct an L_j for each cell. Then the choice of L_j for approximating $u(y)$ is determined by the cell Q_j containing y.

Only the latter procedure is considered in this paper.

Returning back to the case of a general Banach space X, we denote by $\mathscr{L}_{m,N} = \mathscr{L}_{m,N}(X)$ the collection of all libraries $\mathcal{L} = \{L_1, \dots, L_N\}$ containing N affine spaces of dimension at most m. If we fix the values of m and N, then the best performance of a library with these fixed values is

$$d_{m,N}(K) := \inf_{\mathcal{L} \in \ \mathscr{L}_{m,N}} E_{\mathcal{L}}(K). \tag{1.14}$$

We call $d_{m,N}$ the *library width* of K. This definition slightly differs from that introduced in [15] in which the spaces L_j are taken to be linear instead of affine.

Library widths include the two standard approximation concepts of widths and entropy. Recall that if K is a compact set in a Banach space X then its Kolmogorov m width is

$$d_m(K) := d_m(K)_X := \inf_{\dim(Y) = m} \operatorname{dist}(K, Y)_X, \tag{1.15}$$

where the infimum is taken over all linear spaces Y of dimension m. Thus the Kolmogorov m width of K is the smallest error that can be obtained by approximation by linear spaces of dimension m. It follows that we can sandwich the library width $d_{m,1}(K)_X$ between Kolmogorov widths by

$$d_{m+1}(K) \le d_m(K_0) = d_{m,1}(K) \le d_m(K), \tag{1.16}$$

where $K_0 = K - x_0$ for some suitable $x_0 \in X$. At the other extreme,

$$d_{0,2^n}(K) = \varepsilon_n(K), \tag{1.17}$$

where $\varepsilon_n(K)$ is the n-th entropy number of K that is, the smallest number ε such that K can be covered by 2^n balls in X of radius ε.

One of the motivations for using library approximation in the context of parametric PDEs with a small value of m is that the current construction of linear reduced models via greedy algorithms has offline cost that increases exponentially as the dimension of the reduced space increases. This is due to the fact that the greedy algorithm needs to search for the reduced basis elements through a large training set which should in principle resolve the solution manifold $\mathcal{M}$ to the same accuracy ε that is targeted for the reduced basis space V_n. For example, it is known that if the Kolmogorov n width $d_n(\mathcal{M})$ decays like $O(n^{-s})$ for some $s > 0$, then taking $\varepsilon = n^{-s}$, this training set should have cardinality $O(e^{C\varepsilon^{-1/s}})$, or equivalently $O(e^{cn})$, for some fixed constants $C, c > 0$. The resulting offline cost becomes prohibitive as ε is getting small (or n is getting large). The reader can find a detailed analysis of this cost of greedy constructions in [7] or [6]. We should note that it was recently shown in [6] that the offline cost of greedy constructions (under certain model assumptions on the parametric coefficients) can be reduced to polynomial growth in ε by using random training set, provided we are now willing to accept results that hold with high probability. In order not to confuse various issues, we put this aside when going further in this paper.

Because of the offline cost, it may be impossible to build a linear model using a greedy algorithm when the user prescribed error is too small. On the other hand, by choosing m small and an appropriate partitioning $(Q_j)_{j=1,\dots,N}$ for Y, the offline cost is moderate and a nonlinear reduced model may be constructed provided N is not too large. Keeping m small may also be useful in other contexts such as saving in the online cost for the forward problem and numerical savings for state and parameter estimation. In fact the latter is one of our main motivations for nonlinear reduced models.

1.3 Outline of the paper

We begin the next section by giving some general remarks on library approximation. We show that if K is a compact set in a Banach space X whose Kolmogorov n widths decay like n^{-r} for some $r > 0$, then given any target accuracy ε and writing $\varepsilon = n^{-r}$ for a suitable integer n, we have $d_{m,N}(K) \le \varepsilon$ provided $N \ge 2^{c(n-m)}$, with c depending only on r. Thus, this result gives a bound on how many spaces would be needed in the library if we restrict the dimension of the component spaces X_j to be at most m. While quantitative, this estimate is very pessimistic since, as is well known, nonlinear methods are not beneficial for certain compact sets.

The remainder of our paper is directed at using library approximation for reduced models for parametric PDEs. We take $K = \mathcal{M}$ where $\mathcal{M}$ is the solution manifold of a parametric elliptic PDE with affine coefficients (1.7). As already indicated, the library approximation studied in this paper can be viewed as first partitioning the parameter set into N cells Q_j and assigning an affine space

$L_j = u_j + V_j$ with V_j of dimension at most m on each cell. The main issues therefore are how to choose the cells and how to design the spaces V_j. Given a target accuracy ε and a prescribed target m for the dimension of the spaces in the library, we are interested in strategies for generating a good partition of Y into N cells with a bound on the number N of cells needed to guarantee the prescribed accuracy.

In §3, we consider libraries where each of the L_j is generated from a local polynomial expansion with $m+1$ terms. We give a tensor product strategy for subdividing the parameter domain into cells Q_j which are hyperrectangles and finding a polynomial space of dimension $m+1$ associated with each cell. Thus, the reduced model can be viewed as a piecewise polynomial (in y) approximation to $u(y)$. We give bounds on N which are a significant improvement over those in §2 and show how these results can be used to give concrete bounds when specific assumptions are made on the affine representation (1.7).

In §4, we present the results of various numerical tests that confirm our theoretical results. First, we compare the performance (on the entire parameter domain Y) of the two primary linear reduced models, namely polynomial and greedy. These results show that the gain in using greedy algorithms is typically dramatic. Then we implement our numerical methods for partitioning in the case of piecewise polynomial nonlinear models, where our examples show that suitable error can be achieved with a reasonable number of cells provided m is not too small. We then provide a discussion and numerical experiments of nonlinear models based on piecewise polynomials in the setting of data assimilation.

Finally in §5, we conclude with remarks on the possible advantages and disadvantages of library-based reduced models for applications such as online solvers, data assimilation, and parameter estimation. This section also gives us an opportunity to mention several areas where further research is needed for a better understanding of nonlinear model reduction.

2 General remarks on library approximation

We begin by making some general remarks on library approximation. The central issue we address in this section is the size of the library needed to achieve a given target accuracy when we require dimension m of the spaces in the library. The following theorem gives a first, very pessimistic, bound for the size of the library, which we denote by N.

Theorem 2.1. *Let K be a compact set in a Banach space X. If for some $x_0 \in X$ the Kolmogorov widths of $K_0 = K - x_0$ satisfy*

$$d_k(K_0)_X \leq Mk^{-r}, \quad k \geq 1, \tag{2.1}$$

for some $M > 0$, then for any $0 \leq m \leq n$, one has

$$d_{m,N}(K) \leq (1 + 2^{2r})Mn^{-r}, \tag{2.2}$$

provided $N \geq B_r^{n-m}$ with B_r depending only on r. In other words, we can obtain the same accuracy as in (2.1) *by using spaces of the lower dimension m, provided we take N of them.*

Proof: Since $K = K_0 + x_0$ and since the definition $d_{m,N}(K)$ uses libraries of affine spaces, it is sufficient to prove the theorem for $x_0 = 0$ and thus $K_0 = K$.

Let us first note that there is a nested sequence of spaces $X_k \subset X_{k+1}$ with $\dim(X_k) = k$ and

$$\operatorname{dist}(K, X_k)_X \leq 2^{2r}Mk^{-r}, \quad k \geq 1. \tag{2.3}$$

Indeed, from (2.1), there are linear spaces L_j, $j \geq 0$, of dimension 2^j, and

$$\operatorname{dist}(K, L_j)_X \leq M 2^{-jr}.$$

The spaces $Y_j := L_0 + \cdots + L_j$ have dimension n_j with $2^j \leq n_j \leq 2^{j+1}$ and satisfy

$$\operatorname{dist}(K, Y_j)_X \leq M 2^{-jr} = 2^{2r} M 2^{-(j+2)r} \leq 2^{2r} M n_{j+1}^{-r}, \quad j \geq 0. \tag{2.4}$$

Since the spaces Y_j are nested, and $n_0 \leq \ldots \leq n_j \leq \ldots$, we can find functions $\phi_1, \phi_2, \ldots$, such that for each j, the functions $\phi_1, \ldots, \phi_{n_j}$ are a basis for Y_j. The spaces

$$X_0 := \{0\}, \quad X_k := \operatorname{span}\{\phi_1, \ldots, \phi_k\}, \quad k \geq 1,$$

provide such a nested sequence, since for $n_j \leq k \leq n_{j+1}$ we have $Y_j \subset X_k \subset Y_{j+1}$ and

$$\operatorname{dist}(K, X_k)_X \leq \operatorname{dist}(K, Y_j)_X \leq 2^{2r} M n_{j+1}^{-r} \leq 2^{2r} M k^{-r}, \quad k \geq 1.$$

Case 1: We fix m and first consider the case when $n = m + 2^j$ with $j = -1, 0, 1, \ldots$, where for the purposes of this proof we replace 2^{-1} by 0 when $j = -1$. We proceed by induction on j and use the nested spaces X_k defined above. We define $W := X_m$ which is a space of dimension m and for each $j \geq 0$, we further define

$$Z_j := \operatorname{span}\{\phi_{m+1}, \ldots, \phi_{m+2^j}\}, \quad \dim(Z_j) = 2^j, \quad \text{and thus } W + Z_j = X_{m+2^j}.$$

We show by induction that for each $j \geq -1$, there is a set $S_j \subset Z_j$ such that:

(i) the library $\mathcal{L}_j := \{s + W, \ s \in S_j\}$ provides the approximation error

$$E_{\mathcal{L}_j}(K) \leq (1 + 2^{2r}) M [m + 2^j]^{-r}, \quad j \geq -1; \tag{2.5}$$

(ii) for each $j \geq -1$, the cardinality of S_j is

$$\#(S_j) =: N_j \leq (1 + 2^{r+1} R)^{2^{j+1}}, \quad R := 1 + 2^{2r+1}. \tag{2.6}$$

When $j = -1$, we can take the set $S_{-1} = \{0\}$. We obtain the desired error bound because of (2.3) and we know that $N_{-1} = 1$.

Suppose now that we have established (i) and (ii) for a value of j. To advance the induction to $j+1$ we do the following. Let $\hat{X} := X/W$ denote the quotient space of X modulo W with elements $[x] = x + W$, $x \in X$. We equip this space with its usual norm

$$\|[x]\|_{\hat{X}} := \operatorname{dist}(x, W)_X. \tag{2.7}$$

We then have the finite dimensional spaces $\hat{Z}_j := \{[z] : z \in Z_j\}$, $j = 0, 1, \ldots$. For each $z_\ell \in S_j \subset Z_j$, we define

$$B_\ell = B([z_\ell], R_0) := \{[z] \in \hat{Z}_{j+1} : \ \|[z] - [z_\ell]\|_{\hat{X}} \leq R_0\}, \quad R_0 := RM[m + 2^j]^{-r},$$

the ball in $\hat{Z}_{j+1}$ with center $[z_\ell]$ and radius R_0. It is known (see [14], p.63) that for any $\varepsilon > 0$, the covering number $N_\varepsilon(U)$ for the unit ball U in $\hat{Z}_{j+1}$ satisfies

$$N_\varepsilon(U) \leq (1 + 2/\varepsilon)^{2^{j+1}}.$$

We next set $\varepsilon := M[m+2^{j+1}]^{-r}$. It follows that the covering number of B_ℓ satisfies

$$N_\varepsilon(B_\ell) \leq (1+2RM[m+2^j]^{-r}/\varepsilon)^{2^{j+1}} \leq (1+2^{r+1}R)^{2^{j+1}}, \quad \ell = 1,\dots,N_j. \tag{2.8}$$

We now take $S_{j+1} \subset Z_{j+1}$ as a collection $\{s\}$ of representatives of the centers $[s]$ of the totality of all the balls of radius ε needed to cover all of the balls B_ℓ, $\ell = 1,\dots,N_j$, that is

$$\bigcup_{\ell=1}^{N_j} B_\ell \subset \bigcup_{s\in S_{j+1}} B([s],\varepsilon).$$

Clearly,

$$\#(S_{j+1}) \leq N_j(1+2^{r+1}R)^{2^{j+1}} \leq (1+2^{r+1}R)^{2^{j+2}}, \tag{2.9}$$

where we have used the induction hypothesis (ii) in the least inequality. This advances the induction assumption for the bound on $\#(S_j)$.

We now check that the library $\mathcal{L}_{j+1} := \{s+W,\ s\in S_{j+1}\}$ provides the desired approximation error bound. Let $x \in K$. Then, it follows from (2.3) that there is a $z \in Z_{j+1}$ such that

$$\|[x]-[z]\|_{\hat{X}} \leq 2^{2r}M[m+2^{j+1}]^{-r}. \tag{2.10}$$

We also know from our induction hypothesis (i) that there is a $z_\ell \in S_j$,such that

$$\|[x]-[z_\ell]\|_{\hat{X}} \leq (1+2^{2r})M[m+2^j]^{-r}.$$

Hence,

$$\|[z]-[z_\ell]\|_{\hat{X}} \leq \|[x]-[z]\|_{\hat{X}} + \|[x]-[z_\ell]\|_{\hat{X}} \leq (1+2^{2r+1})M[m+2^j]^{-r},$$

and so $[z]$ is in the ball B_ℓ. Therefore, there is an $s \in S_{j+1}$ such that

$$\|[z]-[s]\|_{\hat{X}} \leq M[m+2^{j+1}]^{-r}.$$

Combining this with (2.10), we obtain

$$\|[x]-[s]\|_{\hat{X}} \leq (1+2^{2r})M[m+2^{j+1}]^{-r}. \tag{2.11}$$

This advances our induction hypothesis on the error bound.
Case 2: We consider any n, not necessarily of the form $m+2^j$. For any j such that $m+2^j \geq n$, the library $\mathcal{L}_j$ will provide the error $(1+2^{2r})Mn^{-r}$ because of (2.5). So, we choose j as the smallest integer such that $2^j \geq n-m$. For this value of j, we have $2^{j-1} \leq n-m$ and from (2.6), we obtain the bound

$$N_j \leq (1+2^{r+1}R)^{2^{j+1}} = B_r^{2^{j-1}} \leq B_r^{n-m}, \tag{2.12}$$

with $B_r := (1+2^{r+1}R)^4$. □

Remark 2.2. *We may restate Theorem 2.1 as follows. If*

$$d_k(K_0) \leq Mk^{-r}, \quad k \geq 1,$$

then for any $\varepsilon > 0$ and $m \geq 0$, there exists a library $\mathcal{L}$ of m dimensional affine spaces which approximates K to accuracy ε, and has cardinality

$$N = \#(\mathcal{L}) \leq \exp(\alpha\varepsilon^{-1/r} - \beta m),$$

with $\beta = \ln(B_r)$ and $\alpha = \ln(B_r)\left[M(1+2^{2r})\right]^{1/r}$. In particular, the library widths of K satisfy

$$d_{m,N}(K) \leq \varepsilon, \quad \textit{whenever} \quad N \geq \exp(\alpha\varepsilon^{-1/r} - \beta m).$$

Theorem 2.1 is very pessimistic since it holds for all compact sets K and general Banach spaces X. As we know in other settings, some compact model classes do not benefit from nonlinear approximation. Also, note that in the proof of the theorem, we use the same space W of dimension m for each of the affine spaces L_j, thereby never taking advantage of any local behavior of the set K. In the following sections of this paper, we study library approximation for the purpose of creating a nonlinear model reduction for parametric elliptic PDEs. We exploit known theorems on the smoothness of the mapping $y \mapsto u(y)$ to give explicit non-uniform and anisotropic tensor product partitions of the parameter domain Y into N cells and create a library of affine spaces that achieves a prescribed target error and whose size obeys much better bounds than those given in this section.

3 Piecewise polynomial approximation for parametric PDE

Before beginning our analysis, we first remark on what we can expect as quantitative results. Nonlinear methods are most effective when the target function, in our case u, is not smooth; for example when it has point singularities or singularities on lower dimensional sets, or it is piecewise smooth. For the parameter to solution map $y \mapsto u(y)$ associated to the elliptic equation (1.6) with affine parametrization (1.7), singularities occur when the function $a(y)$ is not strictly positive. The uniform ellipticity assumption (1.9) ensures that the singularities of u are located outside the parameter domain Y. However, as $a_{\min}/a_{\max}$ becomes small, they get closer to the boundary of Y, and the use of nonlinear methods becomes more relevant in those cases.

We shall see that the bounds on the number of cells necessary in a partition generated by the nonlinear method remain modest when a reasonable number of terms m in the polynomial approximation are used on each cell; see Table 1. In the final section of this paper, we discuss the advantages this fact provides for online solvers and state estimation.

3.1 Polynomial approximation error

If $\Lambda \subset \mathcal{F}$ is a finite set of indices, we denote by $\mathcal{P}_\Lambda$ the linear space of all V valued polynomials

$$P(y) = \sum_{\nu \in \Lambda} c_\nu y^\nu, \tag{3.1}$$

where the coefficients c_ν are in V. Here and later we use standard multivariate notation, for example, $y^\nu = y_1^{\nu_1} \cdots$ when ν has finite support. We always assume that the set Λ is a *downward closed (or lower) set, that is,*

$$\nu \in \Lambda \quad \text{and} \quad \mu \leq \nu \implies \mu \in \Lambda, \tag{3.2}$$

where $\mu \leq \nu$ means that $\mu_j \leq \nu_j$ for all j. In particular, the null multi-index is contained in Λ. Once the coefficients c_ν are fixed, each $P(y)$ is in the affine space

$$c_0 + \mathrm{span}\{c_\nu \in V \,:\, \nu \in \Lambda^*\}, \qquad \Lambda^* := \Lambda \setminus \{0\}, \tag{3.3}$$

which has dimension no more than $\#(\Lambda^*) = \#(\Lambda) - 1$. A typical choice for the c_ν are the Taylor coefficients in the expansion (1.3).

There are two types of assumptions on the diffusion coefficient commonly employed when proving results on polynomial approximation to u. The first one is to assume a decay rate for the sequence $(\|\psi_j\|_{L_\infty(D)})_{j\geq 1}$. The second type of assumption (and the one we employ here), described in [1], is to assume a local interaction bound on how the supports of the ψ_j overlap. One could derive bounds similar to those given below in the first setting as well.

We assume throughout this section that $u(y)$ is the solution to (1.6) with diffusion coefficient $a(y)$ given by (1.7) and that there is a positive sequence $(\rho_j)_{j\geq 1}$ such that

$$\kappa := \min_{j\geq 1} \rho_j > 1, \tag{3.4}$$

and

$$\delta := \left\| \frac{\sum_{j\geq 1} \rho_j |\psi_j|}{\bar{a}} \right\|_{L^\infty(D)} < 1. \tag{3.5}$$

The following theorem gives a bound for the error of approximation of u by polynomials from P_Λ.

Theorem 3.1. *Assume that* (3.4) *and* (3.5) *hold with* $(\rho_j^{-1})_{j\geq 1} \in \ell_q(\mathbb{N})$ *for* $0 < q < 2$. *For each* $m \geq 1$, *there is a set* Λ *with* $\#(\Lambda) = m$ *such that the* V *valued polynomial* $P(y) := \sum_{\nu\in\Lambda} t_\nu y^\nu$, $y \in Y$, *satisfies*

$$\sup_{y\in Y} \|u(y) - P(y)\|_V \leq C(\delta, \rho, q) \|(\rho_j^{-1})_{j\geq 1}\|_{\ell_q} m^{-r}, \quad r = 1/q - 1/2, \tag{3.6}$$

where $C(\delta, \rho, q) := C(\rho, q) C_\delta$ *with*

$$C(\rho, q) := \beta^{\frac{1}{q}} \exp\Big(\frac{\beta}{q} \|(\rho_j^{-1})_{j\geq 1}\|_{\ell_q}^q\Big), \quad \beta := -\ln(1 - \kappa^{-q})\kappa^q, \quad C_\delta^2 := \frac{(2-\delta) a_{\max}}{(2 - 2\delta) a_{\min}^3} \|f\|_{V'}^2. \tag{3.7}$$

The set Λ *can be chosen to be a lower set and is derived explicitly in the proof.*

Proof: The proof follows from a general summability result established in [1] together with concrete estimates for the constants given in [4]. For the completeness and clarity of the present paper, we provide the details. We first choose Λ to be the set of indices in $\mathcal{F}$ that correspond to the m largest of the numbers $\rho^{-\nu}$. Ties are handled in such a way that Λ is a lower set, see [4]. Then, for $P(y) := \sum_{\nu\in\Lambda} t_\nu y^\nu$ we have by Hölder's inequality that for any $y \in Y$,

$$\|u(y) - P(y)\|_V \leq \sum_{\nu\notin\Lambda} \|t_\nu\|_V \leq \Big(\sum_{\nu\in\mathcal{F}} \rho^{2\nu} \|t_\nu\|_V^2\Big)^{\frac{1}{2}} \Big(\sum_{\nu\notin\Lambda} \rho^{-2\nu}\Big)^{\frac{1}{2}}. \tag{3.8}$$

From the proof of Theorem 2.2 in [1], we know also that

$$\sum_{\nu\in\mathcal{F}} \rho^{2\nu} \|t_\nu\|_V^2 \leq \frac{(2-\delta)\|\bar{a}\|_{L^\infty(D)}}{(2-2\delta)\inf_{x\in D} \bar{a}(x)^3} \|f\|_{V'}^2 \leq C_\delta^2, \tag{3.9}$$

where C_δ is defined in (3.7). Moreover, we have

$$\sum_{\nu\notin\Lambda}\rho^{-2\nu} = \sum_{\nu\notin\Lambda}\rho^{-\nu(2-q)}\rho^{-\nu q} \le \Big(\sup_{\nu\notin\Lambda}\rho^{-\nu(2-q)}\Big)\sum_{\nu\notin\Lambda}\rho^{-\nu q}. \tag{3.10}$$

We now let $(\gamma_k)_{k\ge1}$ be a non-increasing rearrangement of the sequence $(\rho^{-\nu})_{\nu\in\mathcal{F}}$. We note that $\gamma_1 = \rho^{-0} = 1$ due to the fact that $\rho_1 > 1$ and $(\rho_j)_{j\ge1}$ is non-decreasing. Then we have

$$\sup_{\nu\notin\Lambda}\rho^{-\nu q} = \gamma_{m+1}^q \le m^{-1}\sum_{k=2}^{m+1}\gamma_k^q \le m^{-1}\sum_{k\ge2}\gamma_k^q = m^{-1}\sum_{\nu\ne0}\rho^{-q\nu}, \tag{3.11}$$

and hence

$$\sup_{\nu\notin\Lambda}\rho^{-\nu(2-q)} \le \Big(m^{-1}\sum_{\nu\ne0}\rho^{-q\nu}\Big)^{\frac{2-q}{q}}. \tag{3.12}$$

Using (3.9) and (3.12) with (3.10) in (3.8), we get

$$\begin{aligned}\|u(y)-P(y)\|_V &\le C_\delta\Big(m^{-1}\sum_{\nu\ne0}\rho^{-q\nu}\Big)^{\frac{2-q}{2q}}\Big(\sum_{\nu\notin\Lambda}\rho^{-\nu q}\Big)^{\frac12}\\ &\le C_\delta m^{-\frac1q+\frac12}\Big(\sum_{\nu\ne0}\rho^{-\nu q}\Big)^{\frac1q}. \end{aligned}\tag{3.13}$$

The final step of the proof is giving an upper bound of the term $\sum_{\nu\ne0}\rho^{-q\nu}$. For this, let $\alpha := \kappa^{-q} < 1$, so that $\rho_j^{-q} \le \alpha$ for all $j\ge1$. Now define $\beta\ge1$ so that $1-\alpha = e^{-\beta\alpha}$, i.e., β is the same as defined in (3.7). Then, β depends only on κ, and q, and by the convexity of $e^{-\beta x}$, we have $1-x\ge e^{-\beta x}$ for $0\le x\le\alpha$. It follows that $(1-\rho_j^{-q})^{-1}\le e^{\beta\rho_j^{-q}}$, and therefore

$$\sum_{\nu\ne0}\rho^{-q\nu} = \prod_{j=1}^{\infty}(1-\rho_j^{-q})^{-1} - 1 \le e^{\beta b}-1 \le \beta b e^{\beta b}, \quad b := \|(\rho_j^{-1})_{j\ge1}\|_{\ell_q}^q. \tag{3.14}$$

Taking the qth root in (3.14) and inserting into (3.13) gives (3.6). □

Remark 3.2. *An important point about the above theorem is that the lower set Λ guaranteed in the theorem can be described a priori by choosing the indices corresponding to the n largest of the numbers $\rho^{-\nu}$ with ties handled properly; see also [8] and [4].*

We next want to derive a local version of the last theorem, namely we want to derive an estimate for how well the Taylor series centered at a general point $\bar y\in Y$ approximates u near $\bar y$. Suppose that $Q_\lambda(\bar y)\subset Y$ is a hyperrectangle centered at some $\bar y\in Y$ with sidelength $2\lambda_j$ in direction j, i.e.,

$$Q_\lambda(\bar y) := \{y\in\mathbb{R}^{\mathbb{N}} : \quad |y_j-\bar y_j|\le\lambda_j, \quad j\ge1\}. \tag{3.15}$$

We refer to the sequence $\lambda := (\lambda_j)_{j\ge1}$ as the sidelength vector for this set.

A first local error estimate for the Taylor series at $\bar y$ is given in the following corollary. In preparation for the proof of that corollary, let us note that for $y\in Q_\lambda(\bar y)$, we have

$$a(y) = a(\bar y) + \sum_{j=1}^{\infty}\frac{(y_j-\bar y_j)}{\lambda_j}(\lambda_j\psi_j) = a(\bar y) + \sum_{j=1}^{\infty}\tilde y_j\tilde\psi_j =: \tilde a(\tilde y), \tag{3.16}$$

where $\tilde{y}_j := \frac{y_j - \bar{y}_j}{\lambda_j} \in [-1,1]$ and $\tilde{\psi}_j := \lambda_j \psi_j$. Therefore,

$$u(y) = \tilde{u}(\tilde{y}), \quad y \in Q_\lambda(\bar{y}),$$

with $\tilde{u}(\tilde{y})$ the solution to

$$-\mathrm{div}\,(\tilde{a}(\tilde{y})\nabla \tilde{u}(\tilde{y})) = f, \quad \tilde{y} \in Y, \tag{3.17}$$

in D with Dirichlet homogeneous boundary conditions.

We can now apply Theorem 3.1 to this new problem (3.17) as long as the assumptions of that theorem hold for this new problem.

Corollary 3.3. *Suppose the assumptions of Theorem 3.1 hold for κ and δ as in (3.4) and (3.5). Consider any hyperrectangle $Q := Q_\lambda(\bar{y}) \subset Y$ as in (3.15) with center $\bar{y} \in Y$ and sidelength vector λ. If there is a sequence $(\tilde{\rho}_j)_{j\geq 1}$ (depending on Q) for which*

(i) $\tilde{\rho}_j \geq \kappa$ for $j \geq 1$;

(ii) $\|(\tilde{\rho}_j^{-1})_{j\geq 1}\|_{\ell_q} \leq \|(\rho_j^{-1})_{j\geq 1}\|_{\ell_q}$;

(iii) $\left\| \frac{\sum_{j\geq 1} \tilde{\rho}_j |\tilde{\psi}_j|}{a(\bar{y})} \right\|_{L^\infty(D)} \leq \delta,$

then for each $m \geq 1$, there is a polynomial P (depending on Q) with m terms (whose indices are given by a lower set) such that

$$\sup_{y\in Q} \|u(y) - P(y)\|_V \leq C(\delta,\rho,q)\|(\tilde{\rho}_j^{-1})_{j\geq 1}\|_{\ell_q} m^{-r}, \quad r = 1/q - 1/2, \tag{3.18}$$

where $C(\delta,\rho,q)$ is the constant from Theorem 3.1.

Proof: This follows from Theorem 3.1 applied to the new problem (3.17). We obtain the same constant because of the assumptions (i)-(iii) placed on the sequence $(\tilde{\rho}_j)_{j\geq 1}$.

3.2 An upper bound on the library size

We now turn to the central issue of given m, and a desired accuracy ε, how can we partition the parameter domain Y into a finite number of cells such that u can be approximated to this accuracy by a piecewise polynomial on this partition, where each polynomial has $m+1$ terms? Deriving such a partition and bounding its size requires some preparatory work. Let $C := C(\delta,\rho,q)$ be the constant of Theorem 3.1. We assume without loss of generality that

$$C\|(\rho_j^{-1})_{j\geq 1}\|_{\ell_q}(m+1)^{-r} > \varepsilon, \tag{3.19}$$

since otherwise the parameter domain Y does not need to be partitioned. Namely, from Theorem 3.1, there is a polynomial with $m+1$ terms which approximates u on Y to accuracy ε. Since $(\rho_j^{-1})_{j\geq 1} \in \ell_q(\mathbb{N})$, we define $J := J(\varepsilon, m) \geq 1$ to be the smallest integer such that

$$\sum_{j\geq J+1} \rho_j^{-q} \leq \frac{1}{2} C^{-q}(m+1)^{qr}\varepsilon^q. \tag{3.20}$$

We will see that the directions $J+1, J+2, J+3, \ldots$, contribute at most $\varepsilon/2$ to the total error and we will not need to subdivide in these directions. For the first J directions, the strategy we use distributes the remaining error equally. To that purpose, we define the quantity

$$\sigma^q := \frac{1}{2J} C^{-q}(m+1)^{qr}\varepsilon^q. \tag{3.21}$$

With this notation, we can rewrite (3.19) and (3.20), respectively, as

$$\|(\rho_j^{-1})_{j\geq 1}\|_{\ell_q}^q > 2J\sigma^q, \quad \text{and} \quad \sum_{j\geq J+1} \rho_j^{-q} \leq J\sigma^q. \tag{3.22}$$

We begin with the following lemma.

Lemma 3.4. *Suppose $Q \subset Y$ is a hyperrectangle with center $z = (z_1, \ldots, z_J, 0, 0, \ldots)$ and sidelength vector $\lambda = (\lambda_1, \ldots, \lambda_J, 1, 1, \ldots)$. If*

$$\lambda_j \leq \sigma(\rho_j - |z_j|) \quad j = 1, \ldots, J, \tag{3.23}$$

then here exists a V valued polynomial P_Q with $m+1$ terms such that

$$\|u(y) - P_Q(y)\|_V \leq \varepsilon, \quad y \in Q. \tag{3.24}$$

Proof: We define

$$\tilde{\rho}_j := \begin{cases} \sigma^{-1}, & \text{if } 1 \leq j \leq J, \\ \rho_j, & \text{otherwise}, \end{cases} \tag{3.25}$$

and verify that $(\tilde{\rho}_j)_{j\geq 1}$ satisfies the assumptions (i)-(iii) of Corollary 3.3 for Q.

We start with (i). It follows from the definition (3.4) of κ and from (3.22) that

$$\sigma^q < \frac{1}{2J}\|(\rho_j^{-1})_{j\geq 1}\|_{\ell_q}^q = \frac{1}{2J}\left(\sum_{j=1}^J \rho_j^{-q} + \sum_{j\geq J+1} \rho_j^{-q}\right) \leq \frac{1}{2}\kappa^{-q} + \frac{1}{2}\sigma^q,$$

and so $\sigma^{-1} > \kappa$. Since we already know $\rho_j \geq \kappa$ for all j, this verifies condition (i).

We now focus on (ii). We set $\eta := C^{-1}\varepsilon(m+1)^r$ and use the choice of J in (3.20) to write

$$\|(\tilde{\rho}_j^{-1})_{j\geq 1}\|_{\ell_q}^q = J\sigma^q + \sum_{j\geq J+1} \rho_j^{-q} \leq J\sigma^q + \frac{1}{2}\eta^q = \eta^q. \tag{3.26}$$

Moreover, if we combine (3.26) with (3.22), we obtain

$$\|(\tilde{\rho}_j^{-1})_{j\geq 1}\|_{\ell_q}^q \leq \eta^q < \|(\rho_j^{-1})_{j\geq 1}\|_{\ell_q}^q,$$

and so (ii) holds.

Finally, to prove (iii), recall that $\tilde{\psi}_j = \lambda_j\psi_j$ and therefore from the inequalities (3.25) and (3.23) we have

$$\tilde{\rho}_j|\tilde{\psi}_j| = \tilde{\rho}_j\lambda_j|\psi_j| \leq (\rho_j - |z_j|)|\psi_j|.$$

This gives

$$\left\| \frac{\sum_{j\geq 1} \tilde{\rho}_j |\tilde{\psi_j}|}{a(z)} \right\|_{L^\infty(D)} \leq \left\| \frac{\sum_{j\geq 1} \rho_j |\psi_j| - \sum_{j\geq 1} |z_j||\psi_j|}{\bar{a} - \sum_{j\geq 1} |z_j||\psi_j|} \right\|_{L^\infty(D)} \leq \delta. \tag{3.27}$$

In view of the definition of δ, see (3.5), the last inequality follows from

$$0 \leq \sum_{j\geq 1} |z_j||\psi_j(x)| < \sum_{j\geq 1} \rho_j |\psi_j(x)| \leq \bar{a}(x), \quad x \in D,$$

and the inequality $\left|\frac{\alpha-\beta}{\gamma-\beta}\right| \leq \left|\frac{\alpha}{\gamma}\right|$ which is valid for any $0 \leq \beta < \alpha \leq \gamma$. Thus, (iii) has been established.

We can now use Corollary 3.3 to guaranteed the existence of the polynomial P_Q to complete the proof. □

We are now in position to state the main theorem of this section.

Theorem 3.5. *Let $0 < q < 2$ and $(\rho_j^{-1})_{j\geq 1} \in \ell_q(\mathbb{N})$ be a nondecreasing sequence which satisfies* (3.4) *and* (3.5). *Let $\varepsilon > 0$, $m \geq 0$ and assume that* (3.19) *holds. Then, there exists a tensor product partition of Y into a collection $\mathcal{R}$ of N hyperrectangles such that on each $Q \in \mathcal{R}$ there is a V valued polynomial P_Q with $m+1$ terms such that*

$$\|u(y) - P_Q(y)\|_V \leq \varepsilon, \quad y \in Q. \tag{3.28}$$

Furthermore, if $J := J(\varepsilon, m)$ is as in (3.20), *then the partition is obtained by only subdividing in the first J directions and the number of cells N in this partition satisfies*

$$N \leq \prod_{j=1}^{J} \left(\sigma^{-1} |\ln(1-\rho_j^{-1})| + C(\sigma)\right) \quad \textit{for some } C(\sigma) \in (1,2). \tag{3.29}$$

Proof: To define our tensor product grid, for each $j = 1, \dots, J$, we define how we subdivide $[-1,1]$ into $(2k_j+1)$ intervals

$$I_j^i, \quad -k_j \leq i \leq k_j$$

for the coordinate y_j. Recall that we do not subdivide any of the coordinate axis when $j > J$, i.e., $k_j = 0$ and $I_j^0 = [-1,1]$ when $j > J$. Also, our partition is symmetric and so $I_j^{-i} = -I_j^i$, $i = 1, \dots, k_j$.

We fix $j \in \{1, \dots, J\}$ and describe our partition of $[-1,1]$ into intervals corresponding to the j-th coordinate. Our first interval I_j^0 is centered at $z_j^0 = 0$ and has sidelength $\lambda_j^0 := \sigma\rho_j$ provided this number is less than one. Otherwise, when $\sigma\rho_j \geq 1$, we define $\lambda_j^0 := 1$, and so $k_j = 0$ and our partition consists only of the one interval $I_j^0 = [-1,1]$. Note that since $(\rho_j)_{j\geq 1}$ is nondecreasing, when this happens it also happens for all larger values of j.

Our partition is symmetric with respect to the origin and so we only describe the intervals to the right of the origin. Our next interval I_j^1 has left endpoint the same as the right endpoint of I_j^0, has center z_j^1 and sidelength λ_j^1, where these numbers are defined by the relationship

$$\lambda_j^1 = \sigma(\rho_j - z_j^1). \tag{3.30}$$

The only exception to this definition is when the right endpoint of this interval is larger than one. Then we recenter the interval so its left endpoint is as before and its right endpoint is one. In this case, we would stop the process and k_j would be one.

We continue in this way moving to the right. So, in general, the interval I_j^i will have its left endpoint equal to the right endpoint of I_j^{i-1}, and will have center z_j^i and sidelength λ_j^i which satisfy

$$\lambda_j^i = \sigma(\rho_j - z_j^i) \tag{3.31}$$

except in the case that such a choice would give a right endpoint larger than one in which we rescale. It follows that the interval I_j^i always satisfies

$$\lambda_j^i \leq \sigma(\rho_j - z_j^i), \quad i = 0, 1, \ldots, k_j, \tag{3.32}$$

with equality except for possibly the last interval $I_j^{k_j}$. We give below a bound for k_j which shows this process is finite.

This partitioning gives a tensor product set $\mathcal{R}$ of hyperrectangles Q. In view of the property (3.32), each of the hyperrectangles satisfies the conditions of Lemma 3.4 and therefore the existence of the polynomials P_Q, $Q \in \mathcal{R}$ satisfying the approximation estimate is guaranteed.

It remains to bound the cardinality of $\mathcal{R}$. For this, we bound k_j, $1 \leq j \leq J$, when $k_j \neq 0$. We obtain the bound we want by monitoring the points

$$R^i = z_j^i + \lambda_j^i, \quad i = 0, 1, \ldots, k_j,$$

Each R^i is the right endpoint of I_j^i as long as $0 \leq i < k_j$. Also we know that $R^{k_j} \geq 1$. Relation (3.31) implies that λ_j^i is chosen so that

$$\frac{\lambda_j^i}{\rho_j - R^i + \lambda_j^i} = \sigma.$$

This gives that

$$(1-\sigma)\lambda_j^i = \sigma(\rho_j - R^i).$$

Since $R^i = R^{i-1} + 2\lambda_j^i$, we have

$$(1-\sigma)(R^i - R^{i-1}) = 2\sigma(\rho_j - R^i).$$

We therefore obtain the recursive formula

$$R^i = \frac{1-\sigma}{1+\sigma}R^{i-1} + \frac{2\sigma}{1+\sigma}\rho_j =: \alpha R^{i-1} + b, \quad i = 1, 2, \ldots,$$

where $R^0 = \rho_j\sigma$, $\alpha := \frac{1-\sigma}{1+\sigma}$, $b := \frac{2\sigma}{1+\sigma}\rho_j$. Therefore, we find

$$\begin{aligned} R^i &= \alpha^i R^0 + (1 + \alpha + \ldots + \alpha^{i-1})b = \alpha^i R^0 + \frac{1-\alpha^i}{1-\alpha}b \\ &= \alpha^i \rho_j \sigma + (1-\alpha^i)\rho_j = \rho_j(1 - \alpha^i(1-\sigma)). \end{aligned} \tag{3.33}$$

The iteration will stop at the smallest integer $k = k_j$ such that $R^k \geq 1$. Since $\sigma^{-1} \geq \kappa > 1$, we have $\sigma < 1$ and the iteration will stop at the smallest integer k such that

$$\alpha^k \leq \frac{1-\rho_j^{-1}}{1-\sigma}.$$

Note that $\frac{1-\rho_j^{-1}}{1-\sigma} < 1$ because $\sigma\rho_j < 1$ (otherwise $k_j = 0$ and $I_j^0 = [-1, 1]$). We are looking for the smallest integer k for which

$$k \geq \frac{\ln\left(1-\rho_j^{-1}\right) - \ln(1-\sigma)}{\ln \alpha},$$

which gives

$$k_j = \left\lceil \frac{\ln(1-\rho_j^{-1}) - \ln(1-\sigma)}{\ln \alpha} \right\rceil < \frac{\ln(1-\rho_j^{-1}) - \ln(1-\sigma)}{\ln \alpha} + 1, \quad j = 1, \dots, J.$$

Therefore, we have the bound

$$n_j := 2k_j + 1 \leq 2\frac{\ln\left(1-\rho_j^{-1}\right) - \ln(1-\sigma)}{\ln\left(\frac{1-\sigma}{1+\sigma}\right)} + 3 = 2\frac{\ln\left(1-\rho_j^{-1}\right)}{\ln\left(\frac{1-\sigma}{1+\sigma}\right)} + C(\sigma),$$

where

$$C(\sigma) := -2\frac{\ln(1-\sigma)}{\ln\left(\frac{1-\sigma}{1+\sigma}\right)} + 3 = \frac{\ln\left(\frac{(1-\sigma)}{(1+\sigma)^3}\right)}{\ln\left(\frac{1-\sigma}{1+\sigma}\right)} \in (1, 2). \tag{3.34}$$

Since $\ln(1+x) \geq \frac{2x}{2+x}$ for $x \geq 0$, we obtain

$$\ln\left(\frac{1+\sigma}{1-\sigma}\right) = \ln\left(1 + \frac{2\sigma}{1-\sigma}\right) \geq 2\sigma,$$

and thus $n_j \leq \sigma^{-1}|\ln\left(1-\rho_j^{-1}\right)| + C(\sigma)$, which brings us to the final calculation

$$N = \prod_{j=1}^{J} n_j \leq \prod_{j=1}^{J} \left(\sigma^{-1}|\ln\left(1-\rho_j^{-1}\right)| + C(\sigma)\right), \tag{3.35}$$

which completes the proof. □

Remark 3.6. *It follows from the proof of Theorem 3.5 that a more precise estimate for the number of cells is*

$$N \leq \prod_{j=1}^{J_0} \left(\sigma^{-1}|\ln\left(1-\rho_j^{-1}\right)| + C(\sigma)\right),$$

where $1 \leq J_0 \leq J$ is the largest integer such that $\sigma\rho_{J_0} < 1$. This comes from the fact that $k_j = 0$ for $J_0 < j \leq J$, i.e., we do not subdivide in the directions $J_0 + 1. \dots, J$.

Let us reformulate the above result in terms of library widths. As we have remarked earlier (see (3.3)), a polynomial approximation with $m+1$ terms is naturally associated with an affine space of dimension at most m. We then obtain a library $\mathcal{L} = \cup_{i=1}^{N} L_i$ of affine spaces $L_i = L_i(P_i, Q_i)$,

$$L_i = c_0^i + \operatorname{span}\{c_\nu^i \in V \ : \ \nu \in \Lambda_i, \ \#(\Lambda_i) \le m\}, \quad i = 1, \dots, N,$$

each associated with the polynomial P_i over a hyperrectangle $Q_i \subset Y$,

$$P_i(y) = c_0^i + \sum_{\nu \in \Lambda_i} c_\nu^i y^\nu, \quad y \in Q_i,$$

and cardinality

$$N \le \prod_{j=1}^{J} \left(\sigma^{-1} | \ln \left(1 - \rho_j^{-1} \right) | + C(\sigma) \right), \quad C(\sigma) \in (1, 2).$$

Moreover, since $\sup_{y \in Q_i} \|u(y) - P_i(y)\|_V \le \varepsilon$ for $i = 1, \dots, N$, we have

$$E_{\mathcal{L}}(\mathcal{M}) = \max_{y \in Y} \min_{L \in \mathcal{L}} \operatorname{dist}(u(y), L)_V \le \varepsilon,$$

and therefore

$$d_{m,k}(\mathcal{M}) \le \varepsilon, \quad \text{whenever} \quad k \ge \prod_{j=1}^{J} \left(\sigma^{-1} | \ln \left(1 - \rho_j^{-1} \right) | + C(\sigma) \right).$$

3.3 Examples

To see how how the bounds for N in Theorem 3.5 grow with decreasing ε, we consider the following standard example:

$$\rho_j = M j^s, \quad j \ge 1, \tag{3.36}$$

where $s > 1/2$ is fixed. From our overriding assumption that $\kappa = \rho_1 > 1$, it follows that $M > 1$. We note at the outset that a similar analysis can be done for other growth assumptions on the sequence $(\rho_j)_{j \ge 1}$, e.g., $\rho_j = 1 + M j^s$ with $M > 0$.

Before beginning our analysis, we wish to orient the reader to what type of results we can expect by reflecting on the corresponding results for polynomial approximation. In that case, we know that for each $r < s - 1/2$ we can find V valued polynomials P_n with n terms that satisfy

$$\max_{y \in Y} \|u(y) - P_n(y)\|_V \le C_r n^{-r}, \quad n = 1, 2, \dots. \tag{3.37}$$

This follows from Theorem 3.1 by choosing a value of $q \in (1/s, 2)$ with $r = 1/q - 1/2$. However, we cannot take $r = s - 1/2$ since the constants C_r tend to infinity as $q \to 1/s$. If we are given a target accuracy ε then we would find the minimal number of terms n to reach this accuracy by optimizing over the choice of q. This type of analysis is subtle and done in [4].

We shall obtain similar results for piecewise polynomial approximation where now the main new ingredient is to bound the number of cells that are needed. We fix the desired target accuracy $\varepsilon > 0$ and the value m and use the a priori bound of Theorem 3.5 to see how many hyperrectangles N are needed to guarantee the accuracy ε using piecewise polynomials with $m+1$ terms to approximate u on each rectangle. We can apply Theorem 3.5 for any q that satisfies $1/s < q < 2$. We consider any such q, fix it for the moment, and investigate the size of N needed to achieve the accuracy ε. Throughout the derivation, we let C denote a constant that depends only on q and may change from line to line. Note that $C_0 := C(\delta, \rho, q)$ depends only on q since ρ and δ are fixed.

Since we have

$$\sum_{j \geq J+1} \rho_j^{-q} = M^{-q} \sum_{j \geq J+1} j^{-sq} \leq C J^{1-sq},$$

the condition (3.20) is satisfied if

$$J = C\left(\varepsilon(m+1)^r\right)^{\frac{q}{1-sq}} = C\lambda^{\frac{q}{1-sq}}, \tag{3.38}$$

where

$$\lambda := \varepsilon(m+1)^r, \quad r = r(q) := \frac{1}{q} - \frac{1}{2}.$$

Defining J by (3.38) gives that the value of σ in the theorem is

$$\sigma = 2^{-1/q} C_0^{-1} J^{-1/q} \lambda = C J^{-s}. \tag{3.39}$$

Theorem 3.5 says that we obtain a partition into N hyperrectangular cells such that there is a polynomial with $m+1$ terms on each cell which achieves the desired accuracy ε. It also gives that the number $N = N(q)$ of these cells can be bounded by

$$N \leq \prod_{j=1}^{J} \Big(\sigma^{-1} |\ln\Big(1-\rho_j^{-1}\Big)| + C(\sigma)\Big) < \prod_{j=1}^{J} \Big(\sigma^{-1} |\ln\Big(1-\rho_j^{-1}\Big)| + 2\Big). \tag{3.40}$$

Since each $\rho_j \geq M > 1$, and $|\ln(1-x)| \leq \frac{x}{1-x}$, for $0 < x < 1$, we have

$$|\ln(1-\rho_j^{-1})| \leq (Mj^s - 1)^{-1} \leq (M-1)^{-1} j^{-s}, \quad j = 1, 2, \ldots. \tag{3.41}$$

Placing this into (3.40) gives

$$N \leq \prod_{j=1}^{J} \big((M-1)^{-1}\sigma^{-1} j^{-s} + 2\big) = \prod_{j=1}^{J} \big(CJ^s j^{-s} + 2\big) \leq C^J J^{sJ} [J!]^{-s} \leq e^{(C+s)J} = e^{C\lambda^{\frac{q}{1-sq}}}, \tag{3.42}$$

where the last inequality uses Stirling's formula.

We examine what this bound guarantees for different values of m:

Case $m = 0$: In this case, we are providing the solution manifold $\mathcal{M}$ with an ε approximation net with N elements. Since $\lambda = \varepsilon$ in this case, the bound (3.42) says we can achieve approximation accuracy ε with such a net with

$$N \leq \exp\left\{C\varepsilon^{-\frac{1}{s-1/q}}\right\}$$

elements for any $q \in (1/s, 2)$. The best choice of q in this case is to choose q as close to 2 as possible thereby getting $N \leq e^{C\varepsilon^{-1/\alpha}}$ for any $0 < \alpha < s - 1/2$. Notice that this is in complete agreement with what we know about the entropy of the solution manifold $\mathcal{M}$. Indeed, from Theorem 3.1, we know the Kolmogorov width of $\mathcal{M}$ satisfies

$$d_n(\mathcal{M}) \leq C_r M n^{-r}, \quad 0 < r < s - 1/2, \tag{3.43}$$

where the constants C_r tend to infinity as r gets closer to $s-1/2$. From Carl's inequality we obtain that the ε covering number of $\mathcal{M}$ is bounded by $e^{C\varepsilon^{-1/r}}$ provided that $r < s-1/2$ which is exactly what the above bound on N gives.

Case of general m**:** In this case, the partitioning gives a library of N affine spaces of dimension m that approximate $\mathcal{M}$ to accuracy ε. In order to compare our results on piecewise polynomial approximation with those for polynomial approximation, we suppose a value of $q \in (1/s, 2)$ has been chosen which gives the accuracy $C_r n^{-r}$, $r = r(q) = 1/q - 1/2$ using polynomials. We obtain the same accuracy $\varepsilon := C_r n^{-r}$ using piecewise polynomial with $m+1$ terms and the above estimate says we can do this with

$$N \leq \exp\left\{C\left(\frac{n}{m+1}\right)^{\frac{r}{s-1/q}}\right\} = \exp\left\{C\left(\frac{n}{m+1}\right)^{\alpha}\right\}, \quad \alpha := \frac{1/q-1/2}{s-1/q},$$

cells chosen as in Theorem 3.5. In this estimate, notice that rather than the bound $e^{C(n-m)}$ derived in §2 for general libraries, we now have the bound $e^{C(n/m)^\alpha}$ which gets more favorable as m gets large. Note that we can always get $\alpha = 1$ by taking $q = \frac{4}{2s+1}$, which belongs to the prescribed range $(1/s, 2)$, since $s > 1/2$ by assumption. Moreover, α tends to infinity as $q \to 1/s$ and to 0 as $q \to 2$.

4 Numerical examples

In this section, we present numerical examples to illustrate the performance of the strategy described above for constructing nonlinear reduced models based on partitioning of the parameter domain Y and using piecewise V valued polynomials subordinate to the chosen partition. For our numerical tests, we consider the elliptic equations (1.6) on the domain $D = [0, 1]^2$ with right-hand side $f = 1$ and an affine diffusion of the form

$$a(x, y) := 1 + \sum_{j=1}^{64} y_j c_j \chi_{D_j}(x), \tag{4.1}$$

where $(D_j)_{j=1}^{64}$ is a partition of D into 64 square cells of equal size. The indexing is assigned randomly and has little effect on the numerical results. Thus, the parameter domain $Y = [-1, 1]^{64}$.

We carry out numerical experiments for different sequences $(c_j)_{j=1,\dots,64}$ that depend on the parameters $a_{\min}$ and s, namely

$$c_j = (1 - a_{\min}) j^{-s}, \quad j = 1, 2, \dots, 64, \tag{4.2}$$

where $s \in \{2, 3, 4\}$ and $a_{\min} \in \{0.1, 0.05, 0.01\}$. Notice that $a_{\min}$ is the true minimum of a on $D \times Y$. Given this sequence, we can take

$$\rho_j := \frac{1 - a_{\min}/2}{1 - a_{\min}} j^s, \quad j = 1, 2, \dots, 64, \tag{4.3}$$

and this gives $\delta = 1 - \frac{a_{\min}}{2}$ in (3.5). A small value for $a_{\min}$ corresponds to a reduction in the domain of analyticity of $u(y)$ near the face $y_1 = -1$. So, each numerical experiment corresponds to an assignment of $a_{\min}$ and s.

4.1 Linear reduced models

We begin this section by considering linear reduced models with the goal of understanding how large the dimension of the linear space has to be in order to guarantee a prescribed error ε. We are also interested to see the effect of different choices for the linear space. In all of our numerical experiments we take the target error to be

$$\varepsilon := 10^{-4}.$$

We consider two choices of linear reduced models:

- Taylor polynomial space;
- reduced basis space based on greedily selected snapshots.

We compare the approximations obtained using a Taylor polynomial with n terms and a reduced basis space of dimension n. In particular, we want to see how large n has to be to achieve the target accuracy ε for these two choices.

In the case of a Taylor polynomial space, the approximant $\bar{u}_n$ is given by

$$\bar{u}_n(y) := \bar{t}_0 + \sum_{\nu \in \Lambda_n^*} \bar{t}_\nu y^\nu \in \bar{t}_0 + V_{n-1}(T), \quad V_{n-1}(T) := \operatorname{span}\{\bar{t}_\nu \,:\, \nu \in \Lambda_n^*\}, \tag{4.4}$$

where $\bar{t}_\nu$ is the approximation of t_ν obtained using a finite element solver of high accuracy (much higher accuracy than the target accuracy ε). We consider two methods to generate the lower set Λ_n^* of cardinality $n-1$ which gives the indices ν in (4.4).

The first method, which we refer to as the *a priori method*, orders the $\rho^{-\nu}$, $\nu \in \mathcal{F}$, in decreasing order according to their size. So $\nu^0 := 0$ is the index giving the largest of these numbers, and $\nu^1, \nu^2, \dots$ denote the indices corresponding to the next largest of the $\rho^{-\nu}$. Ties are handled in such a way that $\Lambda_n := \{\nu^0, \nu^1, \dots, \nu^{n-1}\}$ is a lower set, see [4]. We then take $\Lambda_n^* := \Lambda_n \setminus \{\nu^0\}$.

In the second method, here referred to as the *adaptive method*, we use the so-called Algorithm LN (largest neighbor) described in [5] to generate an index set $\tilde{\Lambda}_n$. It begins with $\nu^0 := 0$ and $\tilde{\Lambda}_0 := \{\nu^0\}$. Then, for $k = 0, 1, \dots, n-1$,

$$\tilde{\Lambda}_{k+1} := \tilde{\Lambda}_k \cup \{\nu^k\}, \quad \text{where} \quad \nu^k \in \operatorname*{argmax}_{\nu \in \mathcal{R}_{\tilde{\Lambda}_k}} \|\bar{t}_\nu\|_V. \tag{4.5}$$

Here, $\mathcal{R}_{\tilde{\Lambda}_k}$ denotes the reduced margin of the current lower set $\tilde{\Lambda}_k$, namely

$$\mathcal{R}_{\tilde{\Lambda}_k} := \{\nu \in \mathcal{F} \setminus \tilde{\Lambda}_k :\ \nu - e_j \in \tilde{\Lambda}_k \quad \text{for all } j \text{ with } \nu_j > 0\}.$$

We then take $\Lambda_n^* := \tilde{\Lambda}_n \setminus \{\nu^0\}$.

We compute the error ϵ_n for each of these choices by taking a large number of random (with respect to the uniform distribution) choices[1] of parameters $y \in Y$, as follows. For each choice y, we take an accurate finite element approximation $\bar{u}(y)$ of $u(y)$ as truth. Note that because $\Lambda_n^* \cup \{0\}$ is a lower set, the Taylor coefficients t_ν, $\nu \in \Lambda_n^* \cup \{0\}$, can be found recursively, see equations (3.1) and (3.2) in [5]. We calculate $\|\bar{u}(y) - \bar{u}_n(y)\|_V$ and the error ϵ_n is then computed by maximizing $\|\bar{u}(y) - \bar{u}_n(y)\|_V$ over the random choices of y.

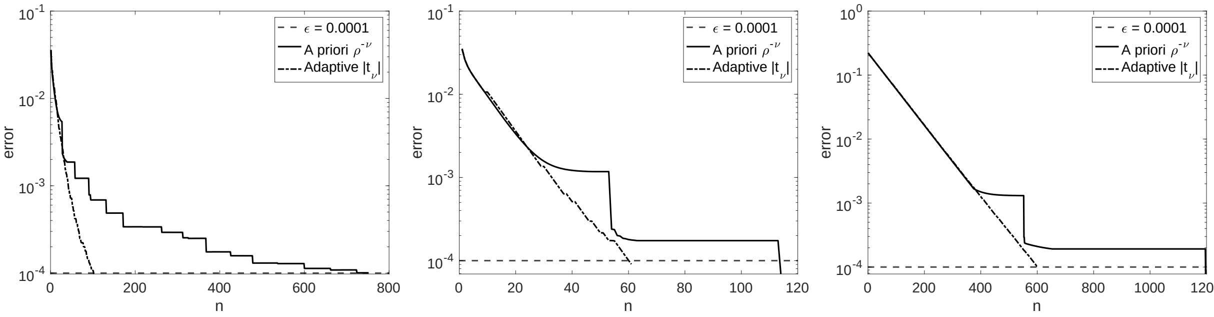


Figure 4.1: Error between $\bar{u}$ and the Taylor polynomial approximation $\bar{u}_n$ versus the number of terms n for both the *a priori method* and the *adaptive method* for constructing Λ_n^*. Left: $s = 2$, $a_{\min} = 0.1$; middle: $s = 4$, $a_{\min} = 0.1$; right: $s = 4$, $a_{\min} = 0.01$.

Figure 4.1 shows a comparison of the errors obtained using the adaptive and the a priori methods to compute the set Λ_n^* as n grows for different values of s and $a_{\min}$. We see that the adaptive method to generate Λ_n^* outperforms the a a priori method, in that the corresponding approximation error is smaller for the adaptive method. This is caused by the fact that $\|\bar{t}_\nu\|_V$ could be much smaller than $\rho^{-\nu}$. On the other hand, the computational cost to find Λ_n^* is greater for the adaptive method. In going further in this section, we always compute the set Λ_n^* for Taylor polynomial indices by using the adaptive method.

We next discuss greedy basis constructions. In this case, the reduced linear space $V_n(G)$ is constructed by starting with the function $\varphi_0 := u(0)$ and then use a particular random weak greedy algorithm [2] to generate the reduced basis functions $\varphi_1, \dots, \varphi_{n-1}$. Each φ_j is a snapshot $\varphi_j = u(y^{(j)})$ of the solution at a judiciously chosen point $y^{(j)} \in Y$. We denote by $\bar{\varphi}_j$ an accurate finite element approximation of φ_j, $j = 0, 1, \dots, n-1$, and we define $V_n(G) := \operatorname{span}\{\bar{\varphi}_0, \bar{\varphi}_1, \dots, \bar{\varphi}_{n-1}\}$. The reduced model is now

$$\bar{u}_n(y) := P_{V_n}(u(y)). \tag{4.6}$$

where P_{V_n} is the Galerkin projection onto V_n(G), namely for a given $y \in Y$, $\bar{u}_n(y) \in V_n(G)$ is the solution of

$$\int\limits_D a(\cdot, y)\nabla\bar{u}_n(y) \cdot \nabla\bar{v}_n = \int\limits_D f\bar{v}_n, \quad \bar{v}_n \in V_n(G).$$

We compute the error for approximating $u(y)$ using random samples of the parameter y in a similar manner to the Taylor case already discussed.

Figure 4.2 gives a comparison of the performance of the greedy basis and the (adaptive) Taylor for different values of s and $a_{\min}$. This graph shows that the greedy basis produces a much more accurate reduced model than the Taylor basis given the same allocation n for the dimension of the reduced space.

[1] In the experiments given the number of random selections of y was 10^3 and using the Mersenne Twister pseudo random generator with seed value 515.

[2] We use a version of the probabilistic weak greedy algorithm given in [6].

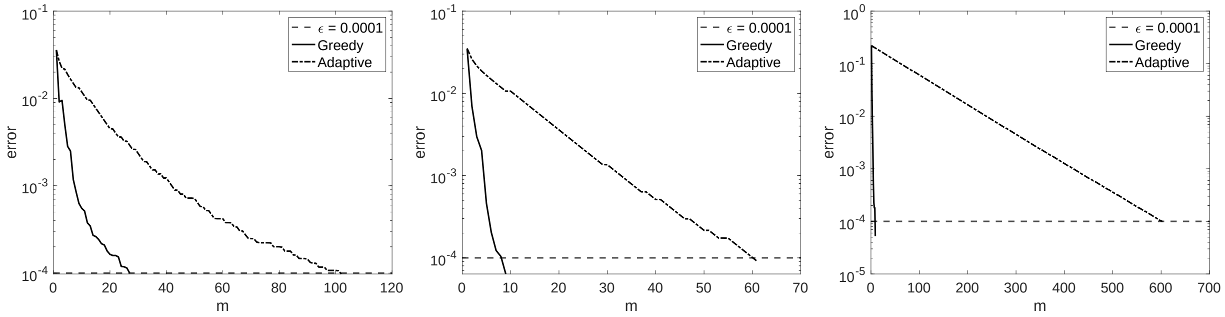


Figure 4.2: Error between $\bar{u}$ and $\bar{u}_n$ versus n for both the (*adaptive*) Taylor and greedy reduced models. Left: $s = 2$, $a_{\min} = 0.1$; middle: $s = 4$, $a_{\min} = 0.1$; right: $s = 4$, $a_{\min} = 0.01$.

4.2 Nonlinear models based on piecewise polynomials

The next set of experiments numerically implements a strategy for generating a nonlinear reduced model based on piecewise polynomials similar to that described in §3. We consider the same diffusion coefficients as above and the same values of s and $a_{\min}$. We again fix a target accuracy $\varepsilon = 10^{-4}$, and a target value of m for the dimension of the polynomial space on each cell of the partition. We will see that it is not always possible to achieve a partition of reasonable size if m is chosen to be too small. This is heuristically clear from the entropy considerations provided in §2 and §3.3.

Our strategy for generating the partitioning of Y into cells is motivated by the theoretical results of §3. However, we make some modifications of this strategy which we now explain. Since in our numerical examples u has singularity near $y = -1$ because $c_j > 0$ for all $j = 1, \dots, 64$, we now grade the partition to be finer near -1 when we refine a coordinate direction. This is in contrast to the theoretical description, which partitions in a symmetric way for each coordinate y_j.

On the other hand, we have found that prescribing ε and m and then implementing the theoretical partitioning strategy actually produces a partition with much better accuracy than ε, and thus we have used too many cells. So instead of viewing the target error and m as the parameters to determine the partition, we introduce a single parameter η to generate a partition. We then select η to give the required accuracy $\varepsilon = 10^{-4}$ and a good control on m and the number of cells N. To be precise, we take $q = 1$ and given $\eta > 0$, we generate a partition as follows.

Construction of the partition for a given η and a non-decreasing sequence $(\rho_j)_{j\geq 1}$:

```
Choose $J \geq 0$ as the smallest integer such that $\sum_{j=J+1}^{64} \rho_j^{-1} \leq \frac{1}{2}\eta$ and set $\sigma = \frac{\eta}{2J}$, $j = 1$;

While $\sigma\rho_j < 1$ do

    $y_j^0 = \frac{1-\sigma\rho_j}{1+\sigma}$, $\lambda_j^0 = \sigma(\rho_j + y_j^0)$, $i = 0$;
    While $y_j^i - \lambda_j^i > -1$
        Increment $i$;
        Compute $\lambda_j^i = \frac{\sigma}{1+\sigma}(\rho_j + y_j^{i-1} - \lambda_j^{i-1}) = \frac{1-\sigma}{1+\sigma}\lambda_j^{i-1}$ and $y_j^i = y_j^{i-1} - \lambda_j^{i-1} - \lambda_j^i$;
    End do
    If $y_j^i - \lambda_j^i < -1$ set $\lambda_j^i = \frac{1}{2}\left(y_j^{i-1} - \lambda_j^{i-1} + 1\right)$ and $y_j^i = \frac{1}{2}\left(y_j^{i-1} - \lambda_j^{i-1} - 1\right)$;
```

```
        Increment $j$;
    End do
    Set $y_l^0 = 0$, $\lambda_l^0 = 1$ for $l = j, ..., J$.
```

The algorithm generates a tensor product partition with cells $Q_\lambda(\bar{y})$ of the form (3.15). For each cell $Q_\lambda(\bar{y})$ from this partition we define a sequence $(\tilde{\rho}_j)_{j\geq 1}$, where

$$\tilde{\rho}_j := \begin{cases} \frac{\rho_j + \bar{y}_j}{\lambda_j}, & \text{when } \sigma\rho_j < 1, \\ \rho_j, & \text{otherwise.} \end{cases}$$

It is easy to check that conditions similar to those in Corollary 3.3 are satisfied. Namely, $\tilde{\rho}_j \geq \kappa$, $j = 1, \dots, 64$, and $\|(\tilde{\rho}_j^{-1})_{j=1}^{64}\|_{\ell_q} \leq \|(\rho_j^{-1})_{j=1}^{64}\|_{\ell_q}$. Moreover, we have

$$\tilde{\delta} := \max_{j=1,\dots,64} \left| \frac{\rho_j c_j + \bar{y}_j c_j}{1 + \bar{y}_j c_j} \right| < 1,$$

since $\rho_j c_j = 1 - a_{\min}/2 < 1$, but not necessarily that $\tilde{\delta} \leq \delta$. However, we can still get the error bound (3.18) of Corollary 3.3, but with constant $C(\delta, \rho, q)$ replaced by the potentially larger constant $C(\tilde{\delta}, \rho, q)$. A uniform error bound can be obtained by taking the constant associated to the largest $\tilde{\delta}$ over all cells in the partition.

Table 1 shows the number of terms m needed in the Taylor expansion on each of the N cells from our partition to meet our error criteria. We see that allowing partitioning can significantly reduce the number m of polynomial terms needed to meet the target accuracy. For example, in the case $N = 1$ (i.e., no partitioning), we need to take $m = 603$ whereas using only ten cells the necessary m is reduced to eight. Note however, that reducing m even further may cause a considerable growth in the number of cells N. Finally, we mention that $J = 1$ in all the examples above.

$a_{min} = 0.1$				$a_{min} = 0.01$			
# of cells	$s = 2$	$s = 3$	$s = 4$	# of cells	$s = 2$	$s = 3$	$s = 4$
$N = 1$	102	68	61	$N = 1$	666	614	603
$N = 3$	29	13	9	$N = 3$	48	30	27
$N = 8$	22	8	5	$N = 10$	24	11	8

Table 1: Number of terms m needed to meet the target accuracy $\varepsilon = 10^{-4}$ on each cell using the piecewise (*adaptive*) Taylor polynomial approximations.

Remark 4.1. *In the above numerical examples, we have not considered the case of using nonlinear models based on piecewise greedy bases. The reason for this is that we do not have an a priori way to generate a good partition of Y into cells when greedy bases rather than polynomial bases are used on each cell. An appropriate strategy would seem to be to do the partitioning in tandem with the local greedy constructions. Strategies for doing this are currently under investigation.*

4.3 State estimation using linear and nonlinear reduced models

As remarked in the introduction, we anticipate that one of the major advantages of using library approximation occurs in the problem of state estimation from data observations. In this section, we recall the state estimation problem and execute several numerical experiments indicating the performance of piecewise polynomial approximations for this problem.

In state estimation, we are given measurements of an unknown state $u(y^*)$ where u is the solution to (1.6) with the model a for the diffusion known to us. We assume that the data is of the form

$$w_j = l_j(u(y^*)), \quad j = 1, \dots, L,$$

where the l_j are linear functionals defined on V. Each linear functional l_j has a Riesz representation

$$l_j(v) = \langle v, \omega_j \rangle_V, \quad j = 1, \dots, L.$$

The functions ω_j, $j = 1, \dots, L$, span a subspace W of V. Without loss of generality, we can assume that the dimension of W is L since otherwise there is redundancy in the measurements.

We want to use these data observations together with the known model a for diffusion in order to construct an approximation $\hat{u}$ to the state $u(y^*)$. Note that y^* and $u(y^*)$ are not necessarily uniquely determined by the measurements. One way of proceeding, as was proposed in [12], is to employ a reduced model based on a linear space V_n to approximate $\mathcal{M}$. The algorithm in [12] constructs an approximation $\hat{u}_n$ to $u(y^*)$ by solving a least squares fit to the data from V_n. This algorithm was shown to be optimal in a certain sense (see [2, 10]) once V_n is chosen. The performance of this algorithm is upper bounded by

$$\|u(y^*) - \hat{u}_n\|_V \le \mu_n \varepsilon_n, \quad \text{where } \varepsilon_n := \operatorname{dist}(\mathcal{M}, V_n)_V. \tag{4.7}$$

Here $\varepsilon_n := \operatorname{dist}(\mathcal{M}, V_n)_V$ and $\mu_n = \mu(W, V_n) \ge 1$ is a certain inf-sup constant which can be interpreted as the reciprocal of the angle between V_n and the space W [3], namely

$$\mu_n = \mu(W, V_n) := \left(\inf_{v \in V_n} \sup_{w \in W} \frac{\langle v, w \rangle_V}{\|v\|_V \|w\|_V} \right)^{-1}.$$

This motivates choosing a nested sequence $V_1 \subset V_2 \subset \cdots$ of spaces with $\dim(V_j) = j$ and selecting a space from this sequence which minimizes the right side of (4.7). Note that while ε_n decreases when increasing n, the constant μ_n increases and is in fact infinite if $n > L$.

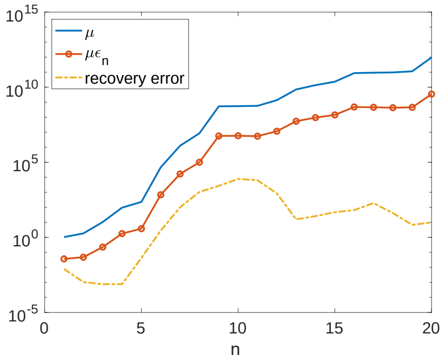
For our numerical experiments in state estimation we use the same models for the diffusion a as described in (4.1)-(4.3). For the measurements, we take linear functionals which emulate point evaluation. Specifically, each l_j is of the form

$$l_j(u) := \int_D u(x) K(x - x_j)\, dx, \quad K(x) := \exp(-\lambda |x|^2), \tag{4.8}$$

where $|x|$ is the Euclidean norm of x and $\lambda = 227.\bar{5}$.

In our numerical experiments, we set $y^* = 0.5384$, but of course operate as if y^* is unknown to us. We take $L = 20$ measurements of the form (4.8), where the centers x_j are chosen at random, applied to the solution $u(\cdot, y^*)$ of (1.6) with a satisfying (4.1)-(4.3) with $s = 4$ and $a_{\min} = 0.1$. We only see these measurements and not the entire function $u(\cdot, y^*)$.

Our first numerical experiment is to compute the behavior of μ_n, the recovery error $\|\bar{u}(y^*) - \hat{u}_n\|_V$ and its upper bound $\mu_n \epsilon_n$, see (4.7), for different choices of V_n, where V_n is the (adaptive) Taylor with n terms and ϵ_n is the approximation error computed as discussed in §4.1. The values obtained for $n = 1, 2, \dots, 20$ when $L = 20$, $s = 4$ and $a_{\min} = 0.1$ are provided in Figure 4.3. The important



n	μ_n	$\mu_n\epsilon_n$	$\|\bar{u}(y^*) - \hat{u}_n\|_V$
5	2.30600×10^2	3.81786×10^0	4.30783×10^{-2}
10	5.48266×10^8	5.82909×10^6	7.82917×10^3
15	2.31818×10^{10}	1.43338×10^8	4.74617×10^1

Figure 4.3: The constant μ_n, the upper bound $\mu_n\epsilon_n$ and the *recovery error* $\|\bar{u}(y^*) - \hat{u}_n\|_V$ for the (*adaptive*) Taylor approximation when $L = 20$, $s = 4$ and $a_{\min} = 0.1$. Left: graphs for $n = 1, 2 \dots, 20$; right: values for $n = 5, 10, 15$.

thing to observe in this figure is that increasing the value of n (in order to improve the approximation error) causes μ to increase greatly and thereby limiting the recovery accuracy. We shall see in the next experiments that this can be circumvented by using piecewise polynomial approximations.

Notice that the dimension n of V_n is limited by $n \leq L$ since otherwise μ_n is infinite. This motivates the use of library approximation with the spaces in the library of small dimension $m \leq L$. We do such a numerical experiment using piecewise Taylor polynomial approximation obtained via the adaptive method. We partition Y into 8 cells. This partition corresponds to only subdividing the first coordinate direction y_1. Each cell gives rise to a "local" value of the inf-sup constant $\mu_m^j := \mu(W, V_m^j)$, $j = 1, 2, ..., 8$, where the V_m^j's are the spaces in the library associated with the partition of Y. Finally, we use $m = 5$ which ensures that the local approximation error satisfies $\epsilon_m^j \leq \varepsilon = 10^{-4}$ for $j = 1, 2, ..., 8$, see Table 1. Figure 4.4 gives the value of μ_m^j, the upper bound $\mu_m^j\epsilon_m^j$ and the *recovery error* $\|\bar{u}(y^*) - \hat{u}_m^j\|_V$, $\hat{u}_m^j \in V_m^j$, for each cell $j = 1, 2, ..., 8$. Notice that the values of μ do not depend on y^*. Also note that the "local" constant μ for the various cells

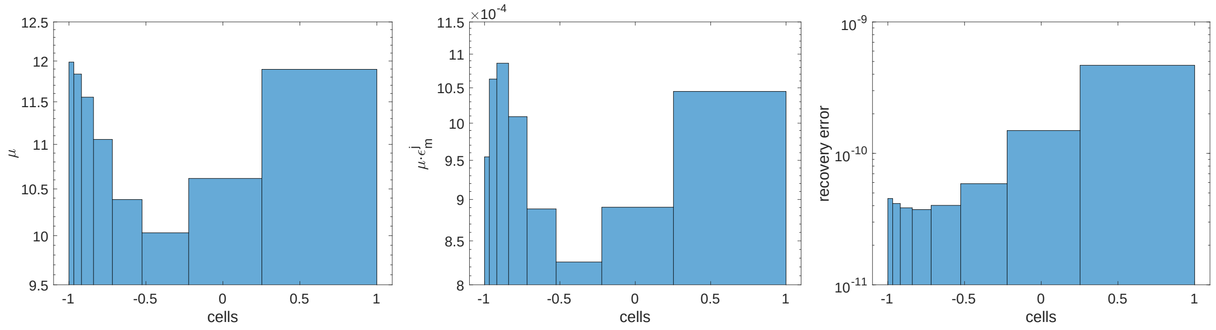


Figure 4.4: Results of the piecewise (*adaptive*) Taylor polynomial approximation on each cell when $L = 20$, $s = 4$ and $a_{\min} = 0.1$. Left: the constant μ_m^j; middle: $\mu_m^j\epsilon_m^j$; right: *recovery error* $\|\bar{u}(y^*) - \hat{u}_m^j\|_V$.

does not exceed 12 while it was about 230 for the one cell case, see Figure 4.3-right. Moreover, we observe that for all the cells, the upper bound $\mu_m^j\epsilon_m^j$ is smaller than 1.1×10^{-3}, which ensures that the *recovery error* (unknown in practice) is less than 1.1×10^{-3}. Note however that we are not providing an algorithm for determining to which cell the parameter y^* is most likely belong to.

5 Conclusions

In this section, we briefly discuss the possible advantages and disadvantages of using nonlinear reduced models in the context of parametric PDEs. We consider only the case of elliptic PDEs (1.6) with affine diffusion coefficients (1.7). We suppose that for the given $(\psi_j)_{j\geq 1}$, there is a nondecreasing sequence $(\rho_j)_{j\geq 1}$ with $\rho_1 > 1$ satisfying (3.5). Quantitative theorems for constructing online solvers with performance guarantees are proven using assumptions on the growth of the sequence $(\rho_j)_{j\geq 1}$. A typical assumption that gives a performance guarantee is that the sequence $(\rho_j^{-1})_{j\geq 1}$ is in $\ell_q(\mathbb{N})$ for some $q < 2$ (see [1, 4]). We assume that we have such a sequence with a fixed value of q. Our discussion is guided by both the theoretical and numerical results of this paper.

5.1 Offline cost for constructing the solver for linear reduced models

Let us first consider the case where our interest is to construct an online solver for the parametric PDE which performs with a guaranteed approximation error ε. There is a distinction in the offline cost of constructing such a solver, depending on whether it is based on Taylor expansions or on a greedy basis expansion.

When using a Taylor polynomial approximation, we need to find a lower set $\Lambda = \Lambda(\varepsilon)$ of indices used in the Taylor polynomial expansion (1.3), where the approximation to the solution belongs to the space $\mathcal{P}_\Lambda$. Recall that we presented two methods for finding such an index set Λ, which we referred to as the *a priori* and the *adaptive* method. The *a priori* method is numerically cheap since it only requires us to sort the $\rho^{-\nu}$ to identify the largest of these numbers (see [4] for one such sorting algorithm). Once the set Λ is identified, the Taylor coefficients $\bar{t}_\nu$ can be computed recursively with finite element solvers as already discussed. The *adaptive* method to build the set Λ may seem more expensive as it requires the computation of all $\bar{t}_\nu$ in the reduced margin of the adaptively constructed monotone set, while only a few may be included in the set Λ_n^*; compare for instance (7.104) and (7.105) in [7]. However, this algorithm is preferred in our numerical experiments presented because it generates sets Λ with eventually smaller cardinality by assessing precisely the magnitude of $\|\bar{t}_\nu\|_V$ instead of using its upper bound $C_u\rho^{-\nu}$ (see [7, Lemma 3.14]).

Consider next the linear reduced model based on the Galerkin projection onto a linear space V_n of dimension n constructed by a weak greedy selection of snapshots from the solution manifold. The advantage of such a greedy construction is that n may be much smaller than the number of terms $\#\Lambda$ used in the Taylor polynomial approximation (see Figure 4.2). Yet, the deficiencies in such greedy algorithms are that the offline cost for the selection of the greedy basis using an ε-net training set grows like $O(\varepsilon^{-c/r}e^{C\varepsilon^{-1/r}})$ (see for instance (8.89) together with (8.108) from [7]) which may be prohibitive for small ε. This of course was one of the main motivations for using nonlinear models in place of linear models.

5.2 Offline cost for constructing a solver using nonlinear reduced models

We discuss next the offline cost in the construction of nonlinear reduced models. Let us first consider reduced models based on piecewise Taylor polynomials. We have given a priori recipes for the tensor product partitioning of Y into cells Q based on the knowledge of the sequence $(\rho_j)_{j\geq 1}$, and thus the main issue is building the appropriate basis for each cell Q of this partition. This requires the computation of the finite element approximation of the appropriate Taylor coefficients on each cell. Note that these computations can be done in parallel. The total cost of this offline construction is governed by the total number N of cells in the partition and the number of terms m used on each cell. In our numerical examples, these constructions were not an issue because the number of cells N was reasonable for moderate values of m.

We have given a priori bounds on the number of cells needed for the partition in §3.2. Recall that if we are in a situation where linear methods (such as polynomial or greedy) give an approximation rate Mn^{-r} then we can guarantee an approximation error $\varepsilon = n^{-r}$ by using piecewise polynomials with m terms and $N \leq e^{C(n/m)^\alpha}$ cells. If we think of the cost of creating a polynomial approximation with m terms to scale like e^{cm}, which we know is the case for greedy constructions, then the cost for constructing the piecewise polynomial is bounded by $e^{C(n/m)^\alpha + cm}$. By choosing $m < n$ appropriately, this is always less than the cost of the approximation without partitioning, which is e^{Cn}. For example, if $\alpha = 1$ then we could choose $m = \sqrt{n}$ and get the total piecewise polynomial cost to be $e^{C\sqrt{n}}$ as compared with the e^{Cn} if we do not partition. In our numerical examples, we have seen that the a priori bounds on the number of cells is quite pessimistic, and we actually get better performance than that predicted by the a priori estimates for the number of cells.

5.3 Online cost for constructing the approximate solution for linear reduced models

If we use a linear reduced model based on Taylor polynomials, then once the index set Λ is found and the Taylor coefficients $\bar{t}_\nu$, $\nu \in \Lambda$, are computed, the reduced model is

$$\bar{u}(y) = \sum_{\nu\in\Lambda} \bar{t}_\nu y^\nu.$$

Thus, given a parameter query, the online cost for the evaluation of $\bar{u}(y)$ is trivial.

If in place of a Taylor polynomial space for the reduced model, we use a greedily generated linear space V of dimension n there are additional online costs. Given a parameter query y one must find the Galerkin projection of $u(y)$ onto V. This entails the inversion of an $n \times n$ dense matrix where the matrix depends on y. In certain cases, such as when the diffusion coefficient is affine, this can be somewhat mitigated by precomputing certain matrices (see the discussion in [7]). Therefore, there is a balancing between having a smaller dimensional reduced model (when compared with the polynomial case) and the additional cost of matrix inversion in an online solver.

Notice also that the accuracy of the online performance given above for reduced models using Taylor polynomials can be improved by using a Galerkin projection onto the polynomial space in place of the plug in formula. However, this projection would also involve an expensive matrix inversion.

5.4 Online cost for constructing the approximate solution for nonlinear reduced models

Building an online solver based on piecewise Taylor polynomial approximations proceeds by building a linear solver for each cell of the partition. An additional step is required to determine which space from the library of spaces should be used for the query y. This only requires the identification of the cell which contains y, and is easily determined from the knowledge of the partition since the cells are hyperrectangles.

5.5 Storage costs

The storage cost for the online solver is dominated by the storage of the basis functions. They are typically large vectors depending on ε, D and f in (1.6). We observe from our numerical experiments that the storage cost is higher for linear reduced models using Taylor polynomials compared to the greedy reduced basis algorithm; see Figure 4.2. Moreover, the costs for Taylor polynomial reduced models and piecewise Taylor polynomial reduced models are quite comparable. For example, from Table 4.2 we realize that for a target accuracy $\varepsilon = 10^{-4}$ and $s = 3$, $a_{\min} = 0.01$, the linear reduced model uses 614 basis functions $\bar{t}_\nu$ while the piecewise Taylor construction has 48 cells with $m = 9$ terms on each cell, and hence requires the storage of 432 vectors.

5.6 Summary

The advantages of a Taylor polynomial based linear reduced model are:

- possible simple identification of the set Λ with no need for optimization or search algorithms;
- fast computation of the online solver $\bar{u}(y)$.

The deficiency in such constructions is that to reach a small target accuracy ε the dimension $m = \#\Lambda$ may be very large and thus affect the offline construction. A large value of m would also affect storage costs.

The advantage of a greedily chosen linear reduced model is that the dimension required for it to reach a target accuracy is typically much smaller than what is required when using Taylor polynomials. The disadvantage is the large offline cost to construct the greedy basis when the required dimension is large, along with the higher cost of executing an online solver. There is, however, a savings in storage because the dimension of the greedy space is small.

A piecewise polynomial nonlinear reduced model has the advantage of being able to achieve a better accuracy than linear reduced models while still taking m small, provided that the number of cells N in the piecewise construction is moderate. In this paper, we have given both a priori bounds on the necessary size of N as well as numerical bounds. Both bounds show the advantage of this approach. The potential deficiency of this approach is a large storage cost if N is large. Our numerical examples suggest that N is considerably smaller than the a priori bounds thereby making this a viable approach when the desired accuracy ε is small.

References

[1] M. Bachmayr, A. Cohen, and G. Migliorati. Sparse polynomial approximation of parametric elliptic PDEs. Part I: affine coefficients. *ESAIM:M2AN*, 51(1):321–339, 2017.

[2] P. Binev, A. Cohen, W. Dahmen, R. DeVore, G. Petrova, and P. Wojtaszczyk. Convergence rates for greedy algorithms in reduced basis methods. *SIAM Journal on Mathematical Analysis*, 43(3):1457–1472, 2011.

[3] P. Binev, A. Cohen, W. Dahmen, R. DeVore, G. Petrova, and P. Wojtaszczyk. Data assimilation in reduced modeling. *SIAM/ASA Journal on Uncertainty Quantification*, 5(1):1–29, 2017.

[4] A. Bonito, R. DeVore, D. Guignard, P. Jantsch, and G. Petrova. Polynomial approximation of anisotropic analytic functions of several variables. *arXiv preprint arXiv:1904.12105 [math.NA]*, 2019.

[5] A. Chkifa, A. Cohen, R. DeVore, and C. Schwab. Sparse adaptive Taylor approximation algorithms for parametric and stochastic elliptic PDEs. *ESAIM: M2AN*, 47(1):253–280, 2013.

[6] A. Cohen, W. Dahmnen, R. DeVore, and J. Nichols. Reduced basis greedy selection using random training sets. *arXiv preprint arXiv:1810.09344 [math.NA]*, 2018.

[7] A. Cohen and R. DeVore. Approximation of high-dimensional parametric PDEs. *Acta Numerica*, 24:1–159, 2015.

[8] A. Cohen and G. Migliorati. Multivariate approximation in downward closed polynomial spaces. In *Contemporary Computational Mathematics - A celebration of the 80th birthday of Ian Sloan*, pages 233–282. Springer, 2018.

[9] R. DeVore. Nonlinear approximation. *Acta Numerica*, 7:51–150, 1998.

[10] R. DeVore, G. Petrova, and P. Wojtaszczyk. Greedy algorithms for reduced bases in Banach spaces. *Constructive Approximation*, 37(3):455–466, 2013.

[11] J.L. Eftang, A.T. Patera, and E.M. Rønquist. An “hp” certified reduced basis method for parametrized elliptic partial differential equations. *SIAM Journal on Scientific Computing*, 32(6):3170–3200, 2010.

[12] Y. Maday, A.T. Patera, J. Penn, and M. Yano. A parameterized-background data-weak approach to variational data assimilation: Formulation, analysis, and application to acoustics. *Int. J. Numer. Meth. Engng*, 102(5):933–965, 2014.

[13] Y. Maday and B. Stamm. Locally adaptive greedy approximations for anisotropic parameter reduced basis spaces. *SIAM Journal on Scientific Computing*, 35(6):A2417–A2441, 2013.

[14] G. Pisier. *The volume of convex bodies and Banach space geometry*, volume 94. Cambridge University Press, 1999.

[15] V. Temlykov. Nonlinear Kolmogorov widths. *Mathematical Notes*, 63(6):785–795, 1998.

[16] Z. Zou, D. Kouri, and W. Aquino. An adaptive local reduced basis method for solving PDEs with uncertain inputs and evaluating risk. *Computer Methods in Applied Mechanics and Engineering*, 345:302–322, 2019.