MOX–Report No. 09/2014

A new algorithm for high-dimensional uncertainty quantification problems based on dimension-adaptive and reduced basis methods

Chen, P.; Quarteroni, A.

MOX, Dipartimento di Matematica "F. Brioschi"
Politecnico di Milano, Via Bonardi 9 - 20133 Milano (Italy)

mox@mate.polimi.it http://mox.polimi.it

A new algorithm for high-dimensional uncertainty quantification problems based on dimension-adaptive and reduced basis methods

PENG CHEN [1] · ALFIO QUARTERONI[1] [2]

Abstract: In this work we develop an adaptive and reduced computational framework based on dimension-adaptive hierarchical approximation and reduced basis method for solving high-dimensional uncertainty quantification (UQ) problems. In order to tackle the computational challenge of "curse-of-dimensionality" commonly faced by these problems, we employ a dimension-adaptive tensor-product algorithm [29] and propose a verified version to enable effective removal of the stagnation phenomenon besides automatically detecting the importance and interaction of different dimensions. To reduce the heavy computational cost of UQ problems modelled by partial differential equations (PDEs), we adopt a weighted reduced basis method [18] and develop an adaptive greedy algorithm in combination of the previous verified algorithm for efficient construction of an accurate reduced basis approximation space. The effectivity, efficiency and accuracy of this computational framework are demonstrated and compared to several other existing techniques by a variety of classical numerical examples.

Keywords: uncertainty quantification, curse-of-dimensionality, generalized sparse grid, hierarchical surpluses, reduced basis method, adaptive greedy algorithm, weighted a posteriori error bound

1 Introduction

While discovering deterministic laws from observation and experimental data remains at the center of computational science and engineering problems, how to identify, quantify and interpret various uncertainties arising from raw data, mathematical models and computational errors become more and more important. Indeed, input data may inevitably be affected by many different uncertainties, for instance computational geometries extracted from noisy images, lack of knowledge for model coefficients and external loadings; another one arises from the increasing interest in more precise statistical assessment of output quantities such as probability distribution, expectation, risk evaluation, sensitivity analysis and so on. Further instances concern the solution of optimal control and parameter identification problems in the presence of uncertainties, data assimilation problems aimed at reducing the uncertainties of the state variable by observational data. These and other requests have lead to the development of statistical and stochastic computational methods in solving, what nowadays falls under the general term of, uncertainty quantifications (UQ) problems.

Many different computational methods have been proposed and developed during the last few decades to solve UQ problems. Among the most widely used is the sampling based Monte Carlo method and its various accelerated versions [25, 20], which are straightforward for implementation

[1]Modelling and Scientific Computing, CMCS, Mathematics Institute of Computational Science and Engineering, MATHICSE, Ecole Polytechnique Fédérale de Lausanne, EPFL, Station 8, CH-1015 Lausanne, Switzerland. Peng Chen (peng.chen@epfl.ch, cpempire@gmail.com) , Alfio Quarteroni (alfio.quarteroni@epfl.ch)

[2]Modellistica e Calcolo Scientifico, MOX, Dipartimento di Matematica F. Brioschi, Politecnico di Milano, P.za Leonardo da Vinci 32, I-20133, Milano, Italy. (on leave)

although it is commonly blamed for slow convergence. A fast convergent method based on the classical idea of projection has been developed under the name of stochastic Galerkin methods, for which different bases of projection can be used such as piecewise finite element and generalized polynomials [30, 74, 3, 66]. However, this method results in a large tensor algebraic system that brings considerable numerical difficulty (ill-posedness) and computational effort (need to design the efficient preconditioners). Another efficient sampling based method, stochastic collocation method [1], has been developed by taking advantage of the easy implementation of Monte Carlo method and the fast convergence of stochastic Galerkin method. In order to alleviate the computational cost, sparse grid techniques [33, 73, 47] are applied to reduce the total number of the nodes or samples. More recently, model order reduction techniques, including the reduced basis method [45, 8, 18] and the proper generalized decomposition method [49, 51], have been developed by expanding the stochastic solution with respect to a few bases that are constructed offline depending on the underlying models. Many more different computational methods have in addition been developed for specific instances of UQ problems, see [72, 50, 2] for reviews.

Unfortunately, all these methods (except the brute-force Monte Carlo method) are affected by a common computational challenge, which is known as "curse-of-dimensionality". When the dimension of the uncertainties becomes high (in the order of 100 and beyond), the number of projection bases or collocation nodes grows exponentially fast such that the computational burden can not be handled by even the most powerful computers. Another computational challenge stems from the fact that when the solution of the underlying model at one sample is expensive, the available computational resource can only afford the full solve at a few tens or hundreds samples, which is far from the required number (in the order of million or beyond) in a high-dimensional space. Any of the two challenges makes it impossible a direct application of the stochastic computational methods introduced above in solving high-dimensional UQ problems.

An opportunity to tackle this "curse-of-dimensionality" is to take advantage of the sparsity – the importance (or sensitivity) of different dimensions and their interaction/combination is very different for the quantities of interest, so that only a limited number of dimensions play an effective role. This role has lead to the development of the weighted function space based quasi Monte Carlo method [20], a priori and a posteriori analysis based anisotropic sparse grid construction [46], (Sobol) decomposition of function based techniques such as ANOVA (analysis of variance) [33, 28, 26], HDMR (high-dimensional model representation) [42], hierarchical surplus based dimension-adaptive generalized sparse grid techniques [9, 29, 33], and so on [7, 6, 44]. The quasi Monte Carlo method improves the convergence rate of the Monte Carlo method (which is $O(1/\sqrt{M})$ when using M randomly chosen samples) by following some digit rules or lattice rules [20] that explore the "weights" of different dimensions when choosing the samples. A faster convergence rate (typically $O((\log(M))^K/M)$ for K dimensional problems) can be achieved in this way. However, when the functions to be approximated feature smoothness and sparsity in the sense that the effective dimensions are much less than the total or nominal dimensions, the quasi Monte Carlo method is still too slow compared to the stochastic Galerkin method or the stochastic collocation method. Smoothness and sparsity have been exploited by anisotropic sparse grid techniques based on either a priori or a posteriori analysis of the convergence rate of the approximate error in each stochastic dimension [46]. This has proved to be more efficient than the isotropic sparse grid in certain test cases. An essential drawback remains for this approach in that the interaction of different dimensions can not effectively be taken into account, leading to either too many useless grid nodes or less accurate approximation for some strongly interacting variables. As the high-dimensional function may be decomposed into a series of low-dimensional additive functions depending on the interaction of different dimensions, the variance based ANOVA (in combination with HDMR) approach has been employed to detect the interactions. Nevertheless, this approach may either be too expensive (more expensive than the original approximation problem based on Lebesgue measure) or not enough accurate (due to arbitrary choice of anchored points based on Dirac measure) and not suitable for high-dimensional interpolation (pointwise evaluation) for stochastic problems with arbitrary probability measure. Another recently developed method under the name of dimension-adaptive tensor-product integration [29] uses a generalized sparse grid construction scheme and employs hierarchical surplus from the construction as error indicators to automatically detect different importance and interaction of different dimensions. Although being essentially equivalent to the anchored ANOVA approach, it is more versatile with different choice of hierarchical surpluses and

suitable for interpolation problems. Still, it is to blame for the drawback of running into stagnation phenomenon, where too early stop of the grid construction in some region occurs before arriving at the desired accuracy of approximation. Another drawback is it use one higher level of grid to assess the error indicators, resulting in a very heavy computational cost.

In this work, we adopt the more versatile dimension-adaptive algorithm based on hierarchical surpluses and generalized sparse grid construction for both integration and interpolation. However, we propose two remedies in addressing the drawbacks and enhancing both its efficiency and accuracy for solving different UQ problems. As for the first drawback of running into stagnation, a balanced greedy algorithm was suggested in [29] and [39], where a purely greedy criteria of choosing the next index by hierarchical surplus for grid construction is balanced by performing the conventional sparse grid construction. However, it is neither possible to choose an optimal balance weight nor feasible to use the same weight throughout the whole grid construction. Alternatively, we propose to carry out a verification procedure in order to get rid of the stagnation phenomenon. The basic idea is that whenever the construction is stopped at some region by meeting certain criteria, we check whether it should be continued by some verification algorithms specific to different dimensional problems. This approach avoids the difficulty in tuning the balanced weight parameter and works efficiently to get out of the stagnation region for grid construction at the appropriate moment.

The verification remedy has not yet been studied in the literature or applied in practice because it needs additional verification samples besides the ones used for assessing hierarchical surpluses in one higher level. This drawback is critical for large-scale UQ problems that already require large computational efforts in solving the underlying PDE model at one sample, as the second computational challenge mentioned before. In order to harness the computational burden, we employ a reduced basis method, which has been used in combination with ANOVA in [36], and develop an adaptive and weighted algorithm in the framework of the verified hierarchical approximation. The rationale of this computational approach is deeply rooted in probability theory: though the random inputs live in a high-dimensional space, the output of interest (statistics of these random inputs) may only lie in a low-dimensional manifold, for instance the arithmetic mean of a large number of independent random variables fulfilling certain conditions (e.g. having finite variance) converge to a (Gaussian) random variable, as guaranteed by the central limit theorem [22]. This fact enables us to construct a reduced bases space with a few number of bases while achieving high accuracy in approximating the high fidelity solution, e.g. finite element solution and the output of interest. Based on this idea and using the reduced basis method for parametrized PDEs [43, 61, 52, 32, 31, 18], we develop an adaptive greedy algorithm in combination with the verified dimension-adaptive hierarchical grid construction procedure to solve high-dimensional UQ problems. In order to take the arbitrary probability measure into account, we use a weighted a posteriori error bound for guiding the selection of the most representative bases [18]. This proves to be more efficient with much less bases in achieving the same approximation accuracy as the a posteriori error bound without incorporating the weight.

By the end, an adaptive and reduced computational framework is developed in efficiently and accurately solving high-dimensional UQ problems that feature sparsity and reducibility. Application of the proposed framework in solving high-dimensional UQ problems based on more general PDE models, such as non-affine, non steady, non-compliant, non-coercive and nonlinear problems can be realized by resorting to specific techniques and computational approaches, e.g. empirical interpolation, primal-dual approach, supermizers enrichment, POD-greedy algorithm and Newton iteration, respectively, which will be summarized in this work. A series of numerical experiments featuring various properties for both functions and PDEs are carried out in demonstrating the efficiency and accuracy of our method and in comparing its computational performance to several other techniques.

The paper is organized as follows. A family of uncertainty quantification problems is introduced in section 2 based on a general formulation. For their numerical solution two computational challenges are identified and briefly illustrated. Section 3 is devoted to the development of the verified dimension-adaptive hierarchical approximation based on generalized sparse grid construction, for which the one dimensional hierarchical interpolation and integration, and a conventional sparse grid construction, is introduced and illustrated at first. Some remarks regarding the computational effectivity, efficiency and accuracy of this method in comparison with some other techniques are provided at the end of this section. In section 4, the adaptive and weighted reduced basis method is presented based on a simple PDE model - a linear, coercive elliptic equation with affine random inputs, where the adaptive greedy

algorithm and weighted a posteriori error bound are developed in details together with the presentation of the offline-online computational decomposition for gaining computational efficiency. Remarks about extensions of the reduced basis method in solving more general PDE models are summarized at the end of the section. A large effort has been devoted to conducting a variety of numerical experiments in section 5, including 10 examples in 6 different topics that offer a rich diversity for demonstrating the accuracy and efficiency of the proposed computational framework and comparing them with several other techniques. In the last section, we close the paper by drawing some conclusions based on the numerical experiments and providing a few further research perspectives for developing and applying the adaptive and reduced computational framework in solving more general high-dimensional UQ problems.

2 Uncertainty quantification (UQ) problems

In this section, we start with defining some basic elements of probability, followed by the presentation of a general formulation of partial differential equations (PDEs) with random inputs. We will denote these equations as "stochastic PDEs" even though this name is traditionally reserved to PDEs with forcing term expressed by a Brownian motion or different kind of noises [71]. Associated with the stochastic PDEs, several uncertainty quantification (UQ) problems largely studied in the literature will be stated in section 2.3. By the end of this section, we identify some common computational challenges in solving the UQ problems.

2.1 Basic notation

Let $(\Omega, \mathfrak{F}, P)$ denote a complete probability space, where Ω is a set of outcomes $\omega \in \Omega$, $\mathfrak{F}$ is a σ-algebra of events and $P : \mathfrak{F} \to [0, 1]$ with $P(\Omega) = 1$ is a probability measure. A real-valued *random variable* is defined as a measurable function $Y : (\Omega, \mathfrak{F}) \to (\mathbb{R}, \mathfrak{B})$, being $\mathfrak{B}$ the Borel σ-algebra on $\mathbb{R}$. The distribution function of a random variable $Y : \Omega \to \Gamma \subset \mathbb{R}$, being Γ the image of Y, is defined as $F_Y : \Gamma \to [0, 1]$ such that with $F_Y(y) = P(\omega \in \Omega : Y(\omega) \leq y)$ and its probability density function $\rho : \Gamma \to \mathbb{R}$ is given by $\rho(y)dy = dF_Y(y)$ provided that the random variable is continuous [22]. For any positive integer $k \in \mathbb{N}_+$, the k-th moment of Y is defined as

$$\mathbb{E}\left[Y^k\right] = \int_\Omega Y^k(\omega) dP(\omega) = \int_\Gamma y^k dF_Y(y) = \int_\Gamma y^k \rho(y) dy. \tag{2.1}$$

Let D be an open and bounded physical domain in $\mathbb{R}^d$ $(d = 1, 2, 3)$ with Lipschitz continuous boundary ∂D. Let $v : D \times \Omega \to \mathbb{R}$ represent a general real-valued *random field*, which is a real-valued random variable defined in Ω for each $x \in D$. We define the product Hilbert space $\mathcal{H}^s(D) := L^2(\Omega) \otimes H^s(D) \equiv L^2_\rho(\Gamma) \otimes H^s(D), s \in \mathbb{R}$ equipped with the norm

$$||v||_{\mathcal{H}^s(D)} := \left(\int_\Omega ||v(\cdot, \omega)||^2_{H^s(D)} dP(\omega) \right)^{1/2} \equiv \left(\int_\Gamma ||v(\cdot, y)||^2_{H^s(D)} \rho(y) dy \right)^{1/2} < \infty, \tag{2.2}$$

where $H^s(D)$ is the Hilbert space of functions defined in the physical domain D [54, 58]. When $s = 0$, we denote $H^0(D) \equiv L^2(D)$, and thus $\mathcal{H}^0(D) \equiv \mathcal{L}^2(D)$ by convention.

2.2 Stochastic partial differential equations

We consider the following stochastic PDE: find $u : \Omega \times D \to \mathbb{R}$ such that the following equations hold almost surely, i.e. for almost every $\omega \in \Omega$

$$\begin{cases} \mathcal{L}(u; x, \omega) & = f(x, \omega) \quad x \in D, \\ \mathcal{B}(u; x) & = g(x) \quad x \in \partial D. \end{cases} \tag{2.3}$$

Here, $\mathcal{L}$ is a differential operator defined in the domain D, $\mathcal{B}$ is a boundary operator defined on the boundary ∂D. f and g represent forcing term and the boundary condition, respectively. The uncertainties of the stochastic PDE may arise from coefficients, geometries, forces and boundary conditions. For simplicity, we only consider the differential operator $\mathcal{L}$ and the forcing term f to be stochastic, while the other sources of uncertainty can be accounted for in a similar way. We assume that the random coefficients in $\mathcal{L}$ and the forcing term f satisfy certain regularity conditions in both the physical space and the probability space so that the stochastic PDE are well posed and the stochastic solution lives in a tensor-product Hilbert space, i.e. $u \in \mathcal{H}^s(D)$ for some $s \in \mathbb{R}$ depending on the regularity of the input data.

Another important assumption, which enables the application of the stochastic computational methods introduced in this work, is the so called "finite dimensional noise assumption" [3, 1]. In short, the uncertainties of the stochastic PDE live in a finite dimensional probability space in the sense that there exist $K \in \mathbb{N}_+$ random variables $Y_k : \Omega \to \mathbb{R}, k = 1, \ldots, K$, such that $\mathcal{L}(u; x, \omega)$ and $f(x, \omega)$ depend on ω only through $Y_1(\omega), \ldots, Y_K(\omega)$, i.e. $\mathcal{L}(u; x, Y_1(\omega), \ldots, Y_K(\omega)) = f(x, Y_1(\omega), \ldots, Y_K(\omega))$. Therefore, $u(x, \omega) = u(x, Y_1(\omega), \ldots, Y_K(\omega))$ and u is measurable with respect to the σ-algebra generated by $Y_1, \ldots, Y_K$, according to Doob-Dynkin Lemma [22]. Let the image of the random variable Y_k be $\Gamma_k \subset \mathbb{R}$ with $Y_k(\omega) = y_k \in \Gamma_k$ for $k = 1, \ldots, K$. Then we define the probability image domain as $\Gamma = \prod_{k=1}^{K} \Gamma_k$ and assume that there exists the joint probability density function $\rho : \Gamma \to \mathbb{R}$. Then the stochastic PDE (2.3) can be recast as a parametric PDE: find $u : \Gamma \times D \to \mathbb{R}$ such that

$$\begin{cases} \mathcal{L}(u; x, y) & = f(x, y) \quad x \in D, \\ \mathcal{B}(u; x) & = g(x) \quad x \in \partial D, \end{cases} \tag{2.4}$$

for almost every parameter $y = (y_1, \ldots, y_K) \in \Gamma$. Therefore, when replacing the stochastic PDEs (2.3) by the equivalent parametric PDEs (2.4), the Lebesgue measure dy is replaced by a probability measure $\rho(y)dy$, e.g. the integral of the solution is given by $\int_\Gamma u\rho(y)dy$.

2.3 Formulation of UQ problems

Associated with the stochastic PDE (2.3), the quantities of interest may be the solution u, the solution associated to a certain physical region or to the boundary, some functional $s : u(y) \to s(y) \equiv s(u(y))$, etc.. Here is a list (far from exhaustive) of uncertainty quantification problems:

1. compute the probability density function or the cumulative distribution function of either u or s [25];
2. evaluate statistical moments, e.g. mean $\mathbb{E}[s]$, variance $\mathbb{V}[s] := \mathbb{E}[s^2] - (\mathbb{E}[s])^2$, etc. [3, 1];
3. perform derivative-based local sensitivity analysis, e.g. compute $du(y)/dy$ or $ds(y)/dy$ [63];
4. perform variance-based global sensitivity analysis, e.g. compute $\mathbb{V}_k[s]/\mathbb{V}[s]$, where $\mathbb{V}_k[s]$ is the variance of s due to the random variable $Y_k, k = 1, \ldots, K$ [10, 15];
5. perform risk analysis, e.g. for a given critical value s_0, compute the failure probability [41, 12]
$$P(\omega \in \Omega : s(\omega) < s_0); \tag{2.5}$$
6. solve stochastic optimal control problems, e.g. the following minimization problem [16, 14, 69]
$$f = \arg\min_{f \in \mathcal{U}_{ad}} \mathcal{J}(u, f) \text{ such that } u \text{ satisfies problem (2.4)}, \tag{2.6}$$
where f is regarded as a deterministic control function living in an admissible space $\mathcal{U}_{ad}$, u_d is a given observation or ideal data, α is a regularization parameter, and the cost functional is
$$\mathcal{J}(u, f) := ||u - u_d||_{\mathcal{H}^s} + \alpha||f||_{\mathcal{L}^2}. \tag{2.7}$$

7. solve an inverse problem, e.g. given experimental data u or s with certain noise η, evaluate the posteriori density ρ_{post} of a random coefficient a of $\mathcal{L}$ based on its priori density ρ_{pr} [5, 64, 75].

From a numerical standpoint, the above UQ problems could be classified as follows: for problems 1, 3, 5, 7, we look for pointwise evaluation of the stochastic solution, i.e. compute $u(y)$ or $s(y)$ at many $y \in \Gamma$; problems 2, 4, 6 require the evaluation of statistical moments. Interpolation techniques are requested for the former class, integration techniques for the latter.

2.4 Computational challenges

Many stochastic computational methods, e.g. the stochastic collocation method [73, 1], face a critical computational challenge: *high dimensionality*, which requires an exponentially increasing number of collocation (for interpolation) or quadrature (for integration) nodes with growing probability dimension K. Figure 2.1 depicts the total number of nodes in tensor product (left) and sparse grid (right) structures with different probability dimensions. The left one reports the results in dimensions 1, 5, 10, 20, 50 and 100 with the number of nodes in each dimension increasing from 1 to 8, from which we can see that the number of nodes easily overpasses the capacity of computational power in relatively high dimensions, e.g. $2^{100} \approx 10^{30}$ nodes are needed for 100 dimensional case with only 2 nodes in each dimension. The results of sparse grid (Smolyak type with Clenshaw–Curtis nodes [73, 47], corresponding to Chebyshev–Gauss–Lobatto nodes in the context of spectral methods [11]) for dimension going up to 200, 500 and 1000 are displayed on the right of Figure 2.1. Compared to tensor product structure, the sparse grid structure considerably reduces the number of nodes, e.g. around 10^6 and 10^9 nodes are needed with 9 nodes in each dimension at sparse grid level 3 for 100 and 1000 dimensional cases. Nevertheless, only tens or hundreds of nodes are affordable in practical engineering problems when a full solve of the underlying PDEs is very expansive. This requirement prevents a direct use of sparse grid techniques for even moderate dimensional problems. This challenge is particularly relevant to UQ problems 6 and 7, namely optimization and inverse problems, for which many full solves (in the order of tens or hundreds) of the underlying PDEs, using some iteration method [54], have to be performed at each of a large number of nodes [13, 14]. A large research effort has been devoted to deal with this computational challenge, e.g. [9, 29, 33, 46, 26, 42, 51, 36, 20].

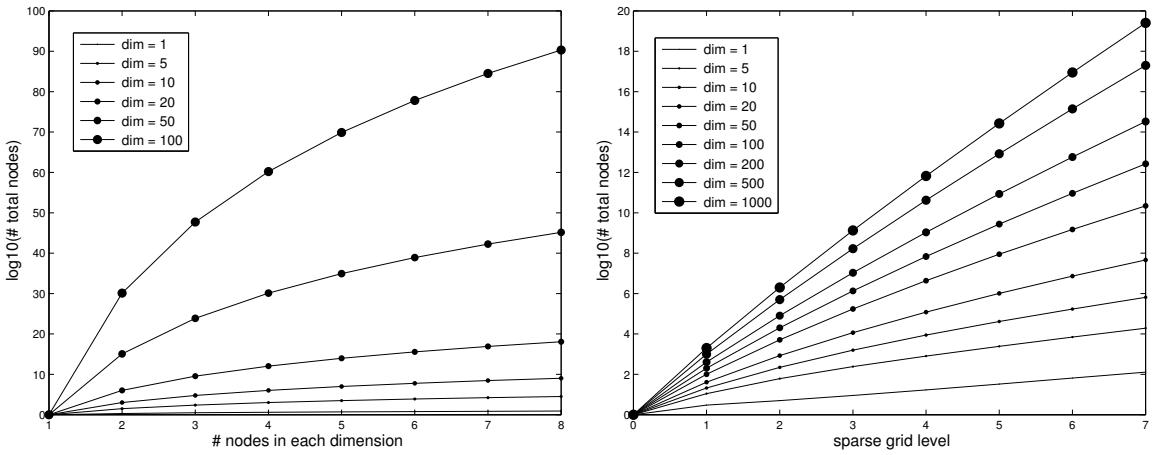

Figure 2.1: Number of collocation (for interpolation) or quadrature (for integration) nodes of tensor product structure (left) and sparse grid structure (right) for different probability dimensions.

Another computational challenge, which we would like to emphasize again, for solving most of the PDE-based UQ problems is that the numerical solution of the underlying PDE model might require a large computational effort: this is e.g. the case of multiscale and/or multiphysics problems. In these circumstances, only a few tens or hundreds of the underlying PDEs can be fully solved, therefore preventing direct application of any method mentioned above in solving high-dimensional

UQ problems, for which a large number (in the order of million and beyond) of PDEs have to be solved in order to evaluate the quantity of interest. This computational challenge is critical for UQ analysis in many practical engineering fields. Research in addressing this challenge in the context of high-dimensional UQ problems is still in its infancy [51].

3 Verified dimension-adaptive hierarchical approximation

In this section, we present the dimension-adaptive tensor-product algorithm for hierarchical approximation of high-dimensional UQ problems based on the work [9, 29, 39]. Our original contribution is to identify the stagnation phenomenon in the hierarchical construction of a generalized sparse grid for this algorithm and propose a verified version of this algorithm in order to cure this undesirable behavior. Suitable error indicators (in particular, a new integration error indicator) are provided for interpolation and integration problems. Some comparisons with several other techniques, e.g. anisotropic sparse grid [46] and variance-based ANOVA (HDMR) [26, 42], are provided at the end of this section.

3.1 Hierarchical interpolation and integration in one dimension

For numerical interpolation of function $s : \Gamma \to \mathbb{R}$ in a one dimensional probability domain $\Gamma \subset \mathbb{R}$, we first pick a series of collocation nodes $y^j \in \Gamma, j = 0, \ldots, m$, ordered such that $y^1 < y^2 < \cdots < y^m$ and for any given $y \in \Gamma$, we approximate the function value $s(y)$ by the interpolation formula

$$s(y) \approx \mathcal{U}s(y) = \sum_{j=1}^{m} s(y^j) l^j(y), \tag{3.1}$$

where $\mathcal{U}$ is an interpolation operator; $l^j, 1 \le j \le m$ are basis functions that, depending on the regularity of the function s with respect to y in Γ, are either piecewise polynomials or global polynomials [57]. For instance, the piecewise linear polynomials most often used in approximating low regularity functions are defined as

$$l^j(y) = \begin{cases} \dfrac{y - y^{j-1}}{y^j - y^{j-1}}, & \text{if } y \in [y^{j-1}, y^j], \quad j = 2, \ldots, m; \\ \dfrac{y^{j+1} - y}{y^{j+1} - y^j}, & \text{if } y \in [y^j, y^{j+1}], \quad j = 1, \ldots, m-1. \end{cases} \tag{3.2}$$

Though converging very slowly (thus requiring a large number of nodes for accurate approximation), these bases lead to uniform convergence when the nodes become dense in the domain Γ. As for the approximation of smooth functions, more suitable are the globally supported polynomials, for instance Lagrange polynomials defined as

$$l^j(y) = \prod_{l=1, l\neq j}^{m} \frac{y - y^l}{y^j - y^l}, \quad j = 1, \ldots, m, \tag{3.3}$$

for a suitable set of nodes such as Gauss quadrature nodes, Chebyshev or Clenshaw–Curtis nodes [57, 70]. For instance, the Clenshaw–Curtis nodes in the interval $[-1, 1]$ are given by

$$y^j = \cos\left(\frac{j-1}{m-1}\pi\right), \quad 1 \le j \le m. \tag{3.4}$$

Let $i \in \mathbb{N}_+$ denote the grid level, Θ^i denote the set of collocation nodes on the grid of level i, with $m(i)$ being the number of nodes on the grid of level i, for instance

$$m(1) = 1; \quad m(i) = 2^{i-1} + 1, \quad i \ge 1. \tag{3.5}$$

We consider nested set of nodes, i.e. $\Theta^i \subset \Theta^{i+1}, i = 1, 2, \ldots, q$ with $q \in \mathbb{N}_+$. In this way, the hierarchical interpolation formula can be written as [9, 39]

$$s(y) \approx \mathcal{U}^q s(y) = \sum_{i=1}^{q} \triangle^i s(y), \tag{3.6}$$

where $\triangle^i$ is the difference of interpolation operators at two successive levels, defined as

$$\triangle^i = \mathcal{U}^i - \mathcal{U}^{i-1}, \quad 1 \le i \le q, \tag{3.7}$$

being $\mathcal{U}^0 = 0$ and $\mathcal{U}^i$ the interpolation operator supported on Θ^i. For notational convenience, let us define $\Theta^i_\triangle = \Theta^i \setminus \Theta^{i-1}, 1 \le i \le q$ with $\Theta^{i-1} := \emptyset$, and reorder the collocation nodes $y^1, \ldots, y^{m(q)}$ in $\Theta^q = \cup_{i=1}^q \Theta^i_\triangle$ level by level in such a way that $y^i_j \in \Theta^i_\triangle, 1 \le i \le q, 1 \le j \le m(i) - m(i-1)$ with $m(0) = 0$. Corresponding to the reordering of the collocation nodes, we denote the basis functions as $l^i_j, 1 \le i \le q, 1 \le j \le m(i) - m(i-1)$. Thanks to the hierarchical structure $\Theta^{i-1} \subset \Theta^i$, $\mathcal{U}^{i-1} s = \mathcal{U}^i \circ \mathcal{U}^{i-1} s$. Moreover, $s(y^i_j) = \mathcal{U}^{i-1} s(y^i_j)$ for $y^i_j \in \Theta^{i-1}$. Therefore, the interpolation operator (3.6) can be rewritten as

$$\mathcal{U}^q s(y) = \sum_{i=1}^{q} \left(\mathcal{U}^i s(y) - \mathcal{U}^i \circ \mathcal{U}^{i-1} s(y) \right) = \sum_{i=1}^{q} \sum_{y^i_j \in \Theta^i_\triangle} \underbrace{\left(s(y^i_j) - \mathcal{U}^{i-1} s(y^i_j) \right)}_{s^i_j} l^i_j(y). \tag{3.8}$$

The real number s^i_j is called *hierarchical surplus* [9], which provides a measure of the interpolation accuracy of the interpolant $\mathcal{U}^{i-1}$ on the successive grid of level i. When this surplus is small, a relatively accurate interpolation is obtained at the corresponding node and grid level.

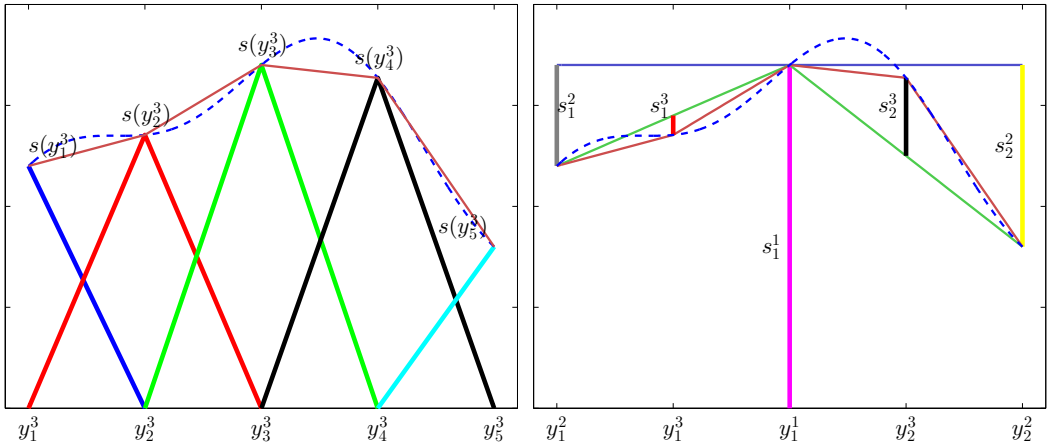

Figure 3.1: Construction of interpolation based on nodal basis (left) and hierarchical basis (right).

The construction of interpolation based on nodal basis (left) and hierarchical basis (right) in the form of piecewise linear polynomials are illustrated in Figure 3.1, from which we can see that the interpolation constructed by the two approaches are equivalent in evaluating function values at any $y \in \Gamma$. However the latter also provides an estimate of interpolation error via hierarchical surpluses. For instance, s^2_1 and s^2_2 are the errors of a constant approximation of the function at nodes y^2_1 and y^2_2, which can provide a rough estimate of the interpolation accuracy.

As for numerical integration in evaluating statistical moments, we can take advantage of the interpolation formula (3.8) and assess the accuracy of integration by hierarchical surplus. For instance,

the expectation of the function s can be computed by

$$\mathbb{E}[s] \approx \mathbb{E}[\mathcal{U}^q s] = \sum_{i=1}^{q} \sum_{y_j^i \in \Theta_\triangle^i} s_j^i w_j^i, \tag{3.9}$$

where the quadrature weights w_j^i are computed by

$$w_j^i = \int_\Gamma l_j^i(y)\rho(y)dy, \quad 1 \le i \le q, 1 \le j \le m(i) - m(i-1) \tag{3.10}$$

using suitable quadrature rules depending on the choice of different collocation nodes [57]. Similarly, the kth ($k \ge 2$) order statistical moments can be evaluated by setting the hierarchical surpluses as $s_j^i = s^k(y_j^i) - \mathcal{U}^{i-1}s^k(y_j^i), 1 \le i \le q, 1 \le j \le m(i) - m(i-1)$.

Based on the hierarchical surplus s_j^i, we may define the interpolation error $\mathcal{E}_i$ and the integration error $\mathcal{E}_e$ as

$$\mathcal{E}_i := \max_{1 \le j \le m(q)-m(q-1)} |s_j^q|, \ \mathcal{E}_e := \sum_{y_j^q \in \Theta_\triangle^q} s_j^q w_j^q. \tag{3.11}$$

There quantities can be used as error indicators in adaptively constructing the interpolation formula (3.8) and integration formula (3.9), respectively. However, one drawback of using the hierarchical surplus as error indicator is that the error may be underestimated where the refinement of the grid has stagnated at an early stage. For instance, in the interpolation constructed from hierarchical basis, the interpolated function values coincide with the true function values at the nodes y_1^3 and y_2^3 as shown in Figure 3.2 in two cases – hierarchical interpolation based on locally supported piecewise linear polynomials and globally supported Lagrange polynomials – so that the hierarchical surplus s_1^3 and s_2^3 become zero, leading to the termination of the adaptive construction of the grid to the next level even the approximation is far from accurate in almost all the region.

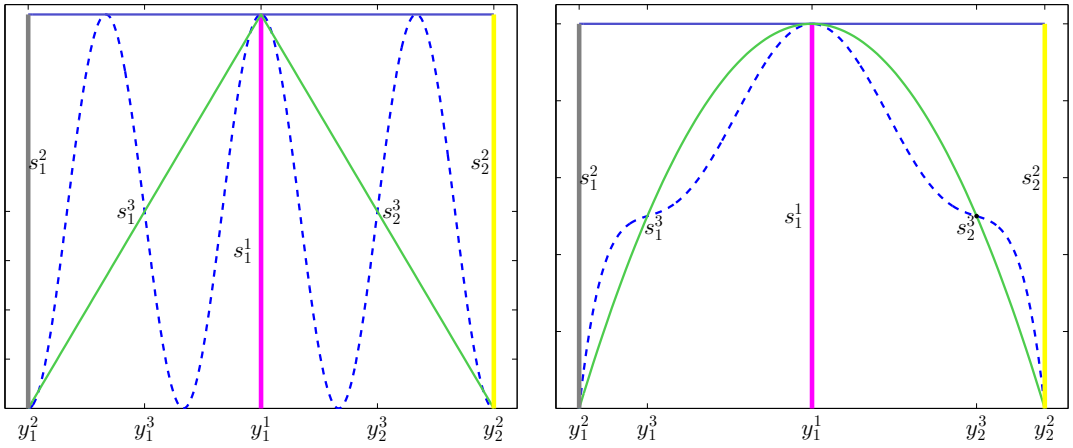

Figure 3.2: Stagnation phenomenons for hierarchical interpolation. Left: piecewise linear polynomials based on equidistant nodes; right: Lagrange polynomials based on Clenshaw–Curtis nodes.

In order to get rid of this stagnation phenomenon, we propose to check the interpolation accuracy (via hierarchical surplus) at the nodes of the next grid level. If the error indicator is larger than the error tolerance, we continue the construction procedure to the next level. Otherwise, we stop. The construction procedure of hierarchical interpolation stopped by satisfying certain error tolerance is summarized in Algorithm 1, which can also be used for hierarchical integration with the interpolation error indicator $\mathcal{E}_i$ replaced by the integration error indicator $\mathcal{E}_e$.

Remark 3.1 *There is the possibility that the error indicator in the next grid level might still be smaller*

than the error tolerance when the approximation is not good enough somewhere, e.g. for continuous functions displaying high oscillation at some very locally supported region that has not been explored by interpolation nodes. In this case, which is also difficult to handle by other interpolation techniques, we may randomly select a certain number of nodes to perform further verification besides using the nodes in the next grid level, expecting that the region can be touched by these nodes with large possibility. This empirical idea needs to be further investigated to balance computational efficiency and accuracy.

Algorithm 1 Verified hierarchical interpolation in one dimension

1: **procedure** INITIALIZATION:
2: specify error tolerance ε_t, type of interpolation bases $l(y)$ and nodes y, specify function $m(i)$;
3: specify maximum level q, set $i = 1$, $\Theta^1 = \{y_j^1, 1 \le j \le m(1)\}$ and evaluate $s_1^1 = s(y_j^1)$;
4: set $\mathcal{E}_i = 2\varepsilon_t$;
5: **end procedure**
6: **procedure** CONSTRUCTION:
7: **while** $\mathcal{E}_i > \varepsilon_t$ and $i \le q$ **do**
8: provide the set of nodes $\Theta_\triangle^i = \{y_j^i, 1 \le j \le m(i) - m(i-1)\}$;
9: for all $y_j^i \in \Theta_\triangle^i$, evaluate function values $s(y_j^i)$ and the interpolation $\mathcal{U}^{i-1}s(y_j^i)$ by (3.8);
10: compute the hierarchical surpluses $s_j^i = s(y_j^i) - \mathcal{U}^{i-1}s(y_j^i)$ and error indicator $\mathcal{E}_i$ by (3.11);
. . . . . .
11: **procedure** VERIFICATION:
12: **if** $\mathcal{E}_i \le \varepsilon_t$ **then**
13: go to the next level $i = i + 1$ and repeat the steps in line 8 - line 10;
14: **end if**
15: **end procedure**
. . . . . .
16: **if** $\mathcal{E}_i \le \varepsilon_t$ **then**
17: **return** .
18: **else**
19: go to the next level $i = i + 1$;
20: **end if**
21: **end while**
22: **end procedure**

3.2 Hierarchical Smolyak sparse grid in multiple dimensions

In multiple dimensional numerical interpolation, when $\Gamma \subset \mathbb{R}^K, K = 2, 3, \ldots$, the univariate interpolation formula (3.6) can be straightforwardly extended as the tensor product interpolation [1]

$$\mathcal{I}_q s(y) := \left(\mathcal{U}_1^q \otimes \cdots \otimes \mathcal{U}_K^q\right) s(y) = \sum_{i_1=1}^{q} \cdots \sum_{i_K=1}^{q} \left(\triangle_1^{i_1} \otimes \cdots \otimes \triangle_K^{i_K}\right) s(y), \tag{3.12}$$

where $\mathcal{U}_k^{q_k}$ and $\triangle_k^{i_k}$ are the univariate interpolation and difference operators in dimension $k = 1, \ldots, K$. Since, as shown in Figure 2.1, the tensor product interpolation needs too many collocation nodes, the Smolyak sparse grid interpolation [67]

$$\mathcal{S}_q s(y) = \sum_{|\mathbf{i}| \le q} \left(\triangle_1^{i_1} \otimes \cdots \otimes \triangle_K^{i_K}\right) s(y) \tag{3.13}$$

is employed to reduce the number of nodes, where the multivariate index $\mathbf{i} = (i_1, \ldots, i_K) \in \mathbb{N}_+^K$ represents the multi-dimensional grid level with *interaction level* $|\mathbf{i}| = i_1 + \cdots + i_K$; $q \ge K$ denotes the *total level* of the isotropic sparse grid. To obtain a hierarchical representation of the sparse grid

interpolation (3.13), we split it as follows

$$\mathcal{S}_q s(y) = \mathcal{S}_{q-1} s(y) + \triangle \mathcal{S}_q s(y), \text{ with } \triangle \mathcal{S}_q s(y) := \sum_{|\mathbf{i}|=q} \left(\triangle_1^{i_1} \otimes \cdots \otimes \triangle_K^{i_K} \right) s(y). \tag{3.14}$$

A more explicit expansion for $\triangle \mathcal{S}_q s(y)$ is

$$\triangle \mathcal{S}_q s(y) = \sum_{|\mathbf{i}|=q} \sum_{\mathbf{j}} \underbrace{\left(s(y_{j_1}^{i_1}, \ldots, y_{j_K}^{i_K}) - \mathcal{S}_{q-1} s(y_{j_1}^{i_1}, \ldots, y_{j_K}^{i_K}) \right)}_{s_{\mathbf{j}}^{\mathbf{i}}} \underbrace{\left(l_{j_1}^{i_1}(y_1) \otimes \cdots \otimes l_{j_K}^{i_K}(y_K) \right)}_{l_{\mathbf{j}}^{\mathbf{i}}}. \tag{3.15}$$

Here, $y_{j_k}^{i_k} \in \Theta_{\triangle}^{i_k}$ is the j_kth node of grid level i_k in dimension $k = 1, \ldots, K$ and $l_{j_k}^{i_k}$ is the corresponding basis function; $s_{\mathbf{j}}^{\mathbf{i}}$ is the hierarchical surplus at node $\mathbf{j}$ of grid level $\mathbf{i}$, which can be used as an error indicator for the construction of adaptive sparse grid. The hierarchical construction of the two dimensional full grid and sparse grid based on Clenshaw–Curtis nodes is illustrated in Figure 3.3 (the size of markers indicates the level of grid), where 1, 4, 8 nodes are added in the 1st, 2nd and 3rd level of sparse grid corresponding to $|\mathbf{i}| = 2, 3, 4$ for the dimension $K = 2$. Note that the sparse grid contains less nodes than the full grid and achieves the same approximation accuracy by taking advantage of the assumption that the interaction level of different dimensions stays small, especially in high-dimensional case. For instance the interpolation (3.12) based on the full grid and (3.13) on the sparse grid in Figure 3.3 can reconstruct exactly any polynomial in the form $y_1^{m(i_1)-1} y_2^{m(i_2)-1}$ such that $i_1 + i_2 \leq 4$. However sparse grid interpolation will produce approximation error when $i_1 + i_2 > 4$, in which case the full grid interpolation is exact as long as $i_1 \leq 3$ and $i_2 \leq 3$. Problems featuring dimensions independent to each other or small interaction level are called *separable* dimensional problems; for them the sparse grid approximation is more favorable. In order to detect the interaction level of different dimensions and enrich the nodes accordingly, the hierarchical surplus $s_{\mathbf{j}}^{\mathbf{i}}$ can be employed directly, as we will see in the next sections.

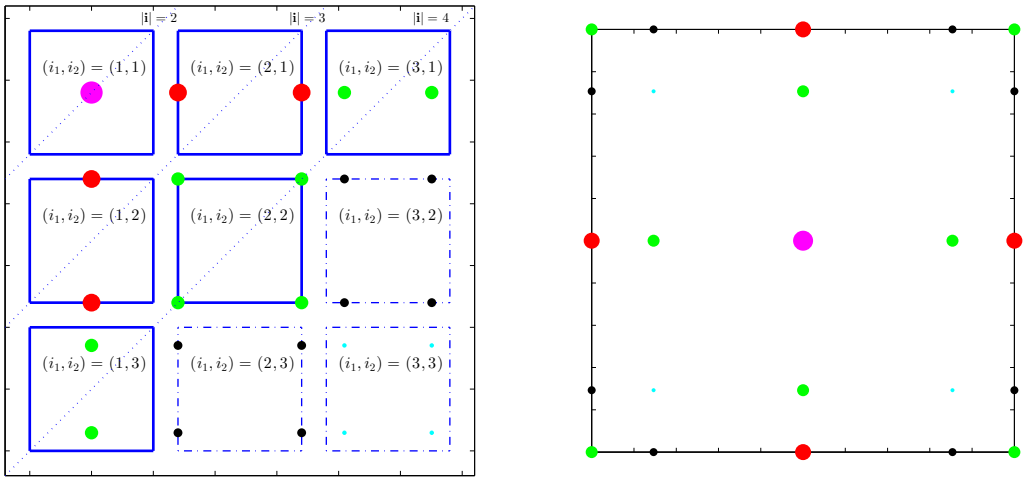

Figure 3.3: Illustration of hierarchical construction of full grid and sparse grid in two dimensions. Left: construction procedure with solid box indicating sparse grid and with the additional dashed box full grid; right: sparse grid (nodes of the first three largest markers in size) and full grid (all nodes).

As for the multivariate numerical integration based on the hierarchical sparse grid interpolation formula (3.13), we obtain the integration formula assembled in a hierarchical form as

$$\mathbb{E}[s] \approx \mathbb{E}[\mathcal{S}_q s] = \sum_{p=K}^{q} \sum_{|\mathbf{i}|=p} \sum_{\mathbf{j}} s_{\mathbf{j}}^{\mathbf{i}} w_{\mathbf{j}}^{\mathbf{i}}, \tag{3.16}$$

where the weight

$$w_{\mathbf{j}}^{\mathbf{i}} = \int_{\Gamma} \left(l_{j_1}^{i_1}(y_1) \otimes \cdots \otimes l_{j_K}^{i_K}(y_K) \right) \rho(y) dy, \tag{3.17}$$

is computed approximately by a suitable quadrature rule depending on the choice of nodes. Provided that the probability density function is separable, i.e. $\rho(y) = \prod_{k=1}^{K} \rho_k(y_k)$, we have

$$w_{\mathbf{j}}^{\mathbf{i}} = \prod_{k=1}^{K} w_{j_k}^{i_k}, \text{ with } w_{j_k}^{i_k} = \int_{\Gamma_k} l_{j_k}^{i_k}(y_k) \rho(y_k) dy_k, 1 \le k \le K, \tag{3.18}$$

which can be precomputed and stored for the sake of computational efficiency. We remark that, when the function s is continuous in Γ, the hierarchical surpluses $s_{\mathbf{j}}^{\mathbf{i}} \to 0$ with $|\mathbf{i}| = q$ as the total approximation level $q \to \infty$ for both interpolation and integration. Therefore, we may estimate the sparse grid interpolation error $\mathcal{E}_i$ and integration error $\mathcal{E}_e$ respectively as

$$\mathcal{E}_i = \max_{|\mathbf{i}|=q, \mathbf{j}} |s_{\mathbf{j}}^{\mathbf{i}}| \text{ and } \mathcal{E}_e = \sum_{|\mathbf{i}|=q} \sum_{\mathbf{j}} s_{\mathbf{j}}^{\mathbf{i}} w_{\mathbf{j}}^{\mathbf{i}}. \tag{3.19}$$

3.3 Dimension adaptation for high-dimensional problems

As we can observe from Figure 2.1, sparse grid introduced in the last section considerably reduces the total number of collocation nodes, making it advantageous to solve moderate (several tens [73, 47]) dimensional approximation problems as well as high (several hundreds or beyond [39]) but separable dimensional problems. However, when the dimensions become too high and the interaction level of different dimensions becomes big, sparse grid techniques are difficult to be directly applied due to computational constraint, e.g. around 10^{12} nodes are needed to approximate 100 dimensional problems with interaction level 7, see Figure 2.1. In this section, we take advantage of the hierarchical surplus and adopt the dimension-adaptive approach [9, 39] to cope with high-dimensional approximation problems. In particular in Algorithm 2, we will propose a high-dimensional verification procedure to deal with possible stagnation phenomena and a new adaptive criterion more suitable for high-dimensional integration problems. Other techniques are also considered for comparison with our proposed approach in a series of remarks.

The sparse grid interpolation based on the difference operator (3.13) is constructed in an isotropic manner due to the restriction $|\mathbf{i}| \le q$. For a more general construction of sparse grid interpolation, we break the isotropic restriction and pose only an *admissibility condition* to satisfy the essential property (3.6) of the hierarchical representation [9, 39]. The set of indices $S \subset \mathbb{N}_+^K$ is called admissible if for each $\mathbf{i} \in S$, the indices $\mathbf{i} - \mathbf{e}_k \in S$ for all $k = 1, \ldots, K$ such that $i_k > 1$. Note that $\mathbf{e}_k \in \{0, 1\}^K$ with the kth element as one and the other elements zero. The sparse grid constructed from an admissible set is called *generalized sparse grid* [9], which includes both the isotropic sparse grid with the index set $S_i := \{\mathbf{i} \in \mathbb{N}_+^K : |\mathbf{i}| \le q\}$ and the full tensor product grid with the index set $S_t := \{\mathbf{i} \in \mathbb{N}_+^K : i_k \le q, 1 \le k \le N\}$. In the admissible index set S_m, being m the cardinality of S_m, we can write the generalized sparse grid interpolation formula (3.13) in a hierarchical way as

$$\mathcal{S}_g s(y) = \sum_{\mathbf{i} \in S_m} \sum_{\mathbf{j}} s_{\mathbf{j}}^{\mathbf{i}} l_{\mathbf{j}}^{\mathbf{i}}. \tag{3.20}$$

Correspondingly, the generalized sparse grid integration formula (3.16) can be written as

$$\mathbb{E}[s] \approx \mathbb{E}[\mathcal{S}_g s] = \sum_{\mathbf{i} \in S_m} \sum_{\mathbf{j}} s_{\mathbf{j}}^{\mathbf{i}} w_{\mathbf{j}}^{\mathbf{i}}. \tag{3.21}$$

At the root level, we set $S_1 = \{\mathbf{1}\}$, in which case the hierarchical surplus $s_{\mathbf{j}}^{\mathbf{i}}$ takes the value of the function s at $y_{\mathbf{j}}^{\mathbf{i}}$. At the next level, we enrich S_1 with the indices of the forward neighborhood of the root index 1, i.e. $S_m = \{\mathbf{1}, \mathbf{1} + \mathbf{e}_k, 1 \le k \le K\}$ with $m = K + 1$ and compute the hierarchical surplus $s_{\mathbf{j}}^{\mathbf{i}}$ for $\mathbf{i} \in S_m \setminus \{\mathbf{1}\}$. Afterwards, the index $\mathbf{i}$ is picked corresponding to the largest error indicator defined

via $s_{\mathbf{j}}^{\mathbf{i}}$ and enrich S_m with the indices from $\{\mathbf{i}+\mathbf{e}_k, 1 \le k \le K\}$ such that S_m remains admissible. Here, we follow [39] to use the averaged hierarchical surplus as the error indicator to pick $\mathbf{i}$

$$\mathbf{i} = \underset{\mathbf{i}\in\mathscr{A}}{\operatorname{argmax}}\, \mathcal{E}_i(\mathbf{i}) \text{ with } \mathcal{E}_i(\mathbf{i}) := \frac{1}{n(\mathbf{i})} \sum_{\mathbf{j}} |s_{\mathbf{j}}^{\mathbf{i}}|, \tag{3.22}$$

where $n(\mathbf{i})$ is the number of nodes added due to the enrichment of the index $\mathbf{i} \in S_m$; $\mathscr{A} \subset S_m$ is the *active index set* collecting all the indices in S_m whose forward neighbors has not been processed. The complementary of $\mathscr{A}$ is called *old index set* with notation $\mathscr{O} = S_m \setminus \mathscr{A}$. After the enrichment, we move the index $\mathbf{i}$ from $\mathscr{A}$ to $\mathscr{O}$ and add the admissible forward neighbors of $\mathbf{i}$ into $\mathscr{A}$ and S_m. Subsequently, we carry out the same procedure to enrich S_m in an adaptive way until satisfying certain stopping criteria, e.g. error tolerance or maximum number of nodes. As for high-dimensional integration, we propose to build the dimension-adaptive sparse grid based on a new error indicator

$$\mathcal{E}_e(\mathbf{i}) := \frac{1}{n(\mathbf{i})} \left| \sum_{\mathbf{j}} s_{\mathbf{j}}^{\mathbf{i}} w_{\mathbf{j}}^{\mathbf{i}} \right|, \tag{3.23}$$

which takes into account three factors: the hierarchical surpluses, the quadrature weights that correspond to arbitrary probability density function and the work contribution by dividing $n(\mathbf{i})$. We remark that the error indicator (3.23) tends to underestimate the integral error since only one index is considered. We provide a more reasonable estimate for the integral error as

$$\mathcal{E}_e(\mathscr{A}) = \left| \sum_{\mathbf{i}\in\mathscr{A}} \sum_{\mathbf{j}} s_{\mathbf{j}}^{\mathbf{i}} w_{\mathbf{j}}^{\mathbf{i}} \right|. \tag{3.24}$$

The construction of the generalized sparse grid in the above procedure not only automatically detects the importance and interaction of different dimensions but also adaptively builds an anisotropic sparse grid without any a priori knowledge or a posteriori processing. However, as in the one dimensional case, stagnation of the adaptive construction might occur at some index $\mathbf{i} \in \mathscr{A}$, thus preventing accurate approximation at an early stage of the hierarchical construction. To overcome this drawback, several algorithms have been proposed in [9, 39] to keep the balance between the purely greedy adaptive construction and a conservative grid construction. For instance, given a weight parameter $w \in [0, 1]$, we add the forward neighbors of the index $\mathbf{i}$, regardless of $\mathcal{E}_i$ or $\mathcal{E}_e$, to the active index set $\mathscr{A}$ as long as [39]

$$\frac{\min_{\mathbf{i}\in\mathscr{A}} |\mathbf{i}|}{\max_{\mathbf{i}\in\mathscr{A}\cup\mathscr{O}} |\mathbf{i}|} \le (1-w), \tag{3.25}$$

where $w = 1$ corresponds to the purely greedy adaptive construction and $w = 0$ the conservative grid construction. Nevertheless, it is not easy to decide what value the weight parameter w should take, leading to either deterioration of the efficiency of the adaptive construction or possible stagnation persisting until a very fine grid has been built. We propose here, as in one dimensional case in Algorithm 1, to perform the verification for each index in the active index set in order to get out of the stagnation set as well as retain the efficiency of the adaptive construction. Our verified dimension-adaptive hierarchical algorithm for interpolation is summarized in Algorithm 2 for high-dimensional interpolation problems. The same algorithm can be adapted for integration by simply replacing the interpolation error indicator $\mathcal{E}_i$ in (3.22) by the integration error indicator $\mathcal{E}_e$ in (3.23). We remark that for function-based high-dimensional interpolation problems, the verified dimension-adaptive hierarchical interpolation algorithm 2 is employed, while for PDE-based interpolation problems, we propose to apply the certified reduced basis method developed in section 4, which produces more accurate approximation results with certification in practice.

As pointed out in [26, 42], in addition to stagnation for the dimension-adaptive hierarchical construction, another drawback is that it involves evaluating the function $s(y)$ at one higher grid level in each dimension in order to assess the error indicator. This is rather costly, especially for high-dimensional uncertainty quantification problems with verification procedure, where the evaluation at

Algorithm 2 Verified dimension-adaptive hierarchical algorithm for interpolation

1: **procedure** INITIALIZATION:
2: specify error tolerance ε_t, types of interpolation bases $l(y)$ and nodes y, specify function $m(i)$;
3: specify maximum number of nodes M, set $\mathbf{i} = \mathbf{1}$, compute $\Theta^{\mathbf{1}}$ and evaluate $s^{\mathbf{1}}_{\mathbf{j}} = s(y^{\mathbf{1}}_{\mathbf{j}}), y^{\mathbf{1}}_{\mathbf{j}} \in \Theta^{\mathbf{1}}$;
4: set $\mathcal{E}_i = 2\varepsilon_t$, $m = \#|\Theta^{\mathbf{1}}|$, $\mathscr{A} = \{\mathbf{1}\}$, $\mathscr{O} = \emptyset$, $S_m = \mathscr{O} \cup \mathscr{A}$;
5: **end procedure**
6: **procedure** CONSTRUCTION:
7: **while** $\mathcal{E}_i > \varepsilon_t$ and $m \leq M$ **do**
8: set $\mathscr{O} = \mathscr{O} \cup \{\mathbf{i}\}$, $\mathscr{A} = \mathscr{A} \setminus \{\mathbf{i}\}$ and enrich $\mathscr{A}$ by the admissible forward neighbors of $\mathbf{i}$;
9: compute the set of nodes $\Theta_\triangle$ different from old nodes at the newly added indices of $\mathscr{A}$;
10: for all $y^{\mathbf{i}}_{\mathbf{j}} \in \Theta_\triangle$, evaluate function values $s(y^{\mathbf{i}}_{\mathbf{j}})$ and the interpolation $\mathcal{S}_g s(y^{\mathbf{i}}_{\mathbf{j}})$ by (3.20);
11: compute the hierarchical surpluses $s^{\mathbf{i}}_{\mathbf{j}} = s(y^{\mathbf{i}}_{\mathbf{j}}) - \mathcal{S}_g s(y^{\mathbf{i}}_{\mathbf{j}})$ and error indicator $\mathcal{E}_i$ by (3.22);
12: increase the number of nodes $m = m + \#|\Theta_\triangle|$, set the total index set $S_m = \mathscr{A} \cup \mathscr{O}$;
. . . . . .
13: **procedure** VERIFICATION:
14: **for** $\mathbf{i}_v \in \mathscr{A}$ **do**
15: **if** $\mathcal{E}_i(\mathbf{i}_v) \leq \varepsilon_t$ **then**
16: set the admissible forward neighbors of $\mathbf{i}_v$ as $\mathscr{A}_v$;
17: compute the set of added nodes $\Theta_\triangle$ for all indices in $\mathscr{A}_v$;
18: repeat lines 10 and 11 with $\mathscr{A}_v$ in (3.22) to get $\mathcal{E}_i$ in $\mathscr{A}_v$;
19: set $\mathscr{O} = \mathscr{O} \cup \{\mathbf{i}_v\}$, $\mathscr{A} = \mathscr{A} \setminus \{\mathbf{i}_v\}$, $\mathcal{E}_i^m = \max_{\mathbf{i}_m \in \mathscr{A}_v} \mathcal{E}_i(\mathbf{i}_m)$;
20: **if** $\mathcal{E}_i^m > \varepsilon_t$ **then**
21: enrich the active set $\mathscr{A} = \mathscr{A} \cup \mathscr{A}_v$ and repeat line 12;
22: **end if**
23: **end if**
24: **end for**
25: **end procedure**
. . . . . .
26: pick the next index $\mathbf{i}$ such that $\mathbf{i} = \text{argmax}_{\mathbf{i} \in \mathscr{A}} \mathcal{E}_i(\mathbf{i})$;
27: **if** $\mathcal{E}_i(\mathbf{i}) \leq \varepsilon_t$ **then**
28: **return** .
29: **end if**
30: **end while**
31: **end procedure**

each of a large number of nodes requires a full solve of the underlying PDE. Fortunately, this computational burden can be considerably alleviated by using the adaptive reduced basis method that will be developed in section 4, where full solve of the underlying PDE model is replaced by a very cheap solve of a reduced model. The corresponding dimension-adaptive approach with verification becomes much more appealing.

3.4 Comparison remarks

In order to take the importance of different dimensions into consideration, an anisotropic sparse grid was proposed in [46] by choosing the index set for the construction of the grid as

$$S_{\boldsymbol{\alpha}} = \left\{ \mathbf{i} \in \mathbb{N}_+^K : \sum_{k=1}^{K} (i_k - 1)\alpha_k \leq q \min_{1 \leq k \leq K} \alpha_k \right\}. \tag{3.26}$$

The multivariate weight $\boldsymbol{\alpha} := (\alpha_1, \ldots, \alpha_K)$ indicates the importance of different dimensions and $q \in \mathbb{N}$ represents the grid level; its choice is a challenging task. The authors suggested two ways to specify $\boldsymbol{\alpha}$ in [46]. In those (simple) cases where a priori estimate for the Lagrange interpolation error exist,

e.g.

$$\sup_{y_k \in \Gamma_k} |s(y_k) - \mathcal{U}_k s(y_k)| \leq C_k e^{-2m_k g(k)}, \quad 1 \leq k \leq K, \tag{3.27}$$

being $\mathcal{U}_k$ the Lagrange interpolation operator and m_k the number of interpolation nodes in dimension k, the weights can be set as $\alpha_k = g(k), 1 \leq k \leq K$. An alternative way to estimate this weight is to perform a posteriori analysis by computing the outputs of interest at a series of collocation nodes and fitting the convergence rate in each dimension. Nevertheless, a posteriori estimate based on error fitting in each dimension can not identify the interaction effect among different dimensions and thus may lead to either not efficient anisotropic sparse grid construction or not accurate approximation. Moreover, the interpolation error may not decay exponentially with respect to the number of nodes for non smooth problems, and no general rule has been proposed for estimating the weight in these circumstances. In comparison, the dimension-adaptive construction of the sparse grid approximation based on hierarchical surpluses does not need to estimate the weights. Instead, it can automatically detect the weight as well as the interaction level among different dimensions as a byproduct of the construction procedure [39].

Another technique to deal with high-dimensional approximation problems is based on ANOVA or HDMR, where the output of interest s can be decomposed into a series of additive functions (in total 2^K) incorporating all the 2^K possible interactions of different dimensions [33], written as

$$s(y) = s_0 + \sum_{1 \leq k_1 \leq K} s_{k_1}(y_{k_1}) + \sum_{1 \leq k_1 < k_2 \leq K} s_{k_1,k_2}(y_{k_1}, y_{k_2}) + \cdots + s_{k_1,\ldots,k_K}(y_{k_1}, \ldots, y_{k_K}), \tag{3.28}$$

with

$$s_0 = \int_\Gamma s(y) d\mu(y), s_{k_1} = \int_{\Gamma^*_{k_1}} s(y) d\mu(y^*_{k_1}) - s_0, s_{k_1,k_2} = \int_{\Gamma^*_{k_1,k_2}} s(y) d\mu(y^*_{k_1,k_2}) - s_0 - s_k, \ldots, \tag{3.29}$$

with $y^*_{k_1} \in \Gamma^*_{k_1}$ in $K-1$ dimensional probability domain except Γ_{k_1}, $y^*_{k_1,k_2} \in \Gamma^*_{k_1,k_2}$ in $K-2$ dimensional probability domain except $\Gamma_{k_1} \times \Gamma_{k_2}$, and so on. Moreover, the variance of the function s admits the same expansion as in (3.28). It is known that there are only a few functions involving a limited number of dimensions play the majority role measured by variance when the function s displays distinctive importance and interaction in different dimensions [33]. Therefore, the high-dimensional approximation problem can be approximated by a series of low-dimensional approximation problems, leading to the development of ANOVA (HDMR) based dimension-adaptive algorithms [37, 24, 26, 42]. However, when the measure μ is the Lebesgue measure, high-dimensional integration has to be carried out in order to evaluate $s_0, s_{k_1}, \ldots$. Alternatively, when μ is a Dirac measure at some anchor point $\bar{y} \in \Gamma$, the expansion (3.28) takes the name of anchored-ANOVA [26] (or cut-HDMR [42]) expansion, which can substantially reduce the computational effort. However, there is no general rule to pick the anchor point, which is critical for accurate approximation and easily results in large error as pointed out in [68]. A single point - centroid of the lowest dimensional tensorial Gaussian quadrature - was suggested as the anchor point in [28]; improvement was also made in [34] by using a screening method, basically selecting several anchor points and taking the average in order to enhance the robustness, which might still not be satisfactory as our numerical examples in section 5 will reveal. Moreover, these variance-based techniques are primarily developed for solving integration problems, which may not be suitable when dealing with pointwise interpolation problems. In contrast, these drawbacks are not faced by the verified dimension-adaptive hierarchical Algorithm 2 that can be used for both high-dimensional interpolation and integration by choosing different error indicators. As a matter of fact, the hierarchical grid construction Algorithm 2 governed by different error indicators plays an equivalent role as automatically decomposing the targeted function into a series of additive functions involving limited dimensions indicated by the interaction of grid level among different dimensions, as demonstrated in the numerical experiments in section 5.

4 Adaptive and weighted reduced basis method

As mentioned in section 2.4, solving PDE-based uncertainty quantification problems faces another critical computational challenge when the underlying PDEs are very difficult to solve. In this circumstance, none of the computational techniques presented in section 3 can be directly applied to deal with high-dimensional UQ problems. In order to tackle this difficulty, we exploit the property that the outputs of interest of the underlying PDEs may live in low-dimensional manifold even though the random inputs are from high-dimensional space. This property, which is known as reducibility, is quite common in practice and is essentially supported by *central limit theorem* and *law of large numbers* in the core of probability theory [22]. In this section, we develop an adaptive and weighted reduced basis method in combination with the hierarchical approximation to efficiently solve high-dimensional UQ problems. Applications to more general PDE models are also provided.

4.1 Reduced basis method

Reduced basis method was initially introduced for structural analysis [48] and recently has undergone vast development in theories [43, 61, 52, 32, 31, 18] and in many engineering applications [55, 19, 56, 40, 59, 12, 14]. Let us present its basic formulation based on the following linear elliptic PDE:

$$-\nabla(a(x,y)\nabla u) = f(x,y) \quad (x,y) \in D \times \Gamma, \tag{4.1}$$

where a and f are random fields standing for the positive diffusion coefficient and the forcing term, respectively. The elliptic PDE (4.1) is closed by homogeneous Dirichlet boundary condition $u = 0$ on ∂D for simplicity. In most practical stochastic modelling and statistical analysis [3, 27, 1, 49], the random fields often admit the following finite affine decomposition:

$$a(x,y) = \sum_{q=1}^{Q_a} \Theta_q^a(y) a_q(x) \text{ and } f(x,y) = \sum_{q=1}^{Q_f} \Theta_q^f(y) f_q(x), \tag{4.2}$$

where Q_a, Q_f are the number of affine terms, Θ_q^a, Θ_q^f are random functions in the probability space and a_q, f_q are deterministic functions in the physical space. In fact, any random field with finite second moment can be decomposed into a finite number of affine terms by the truncated Karhunen-Loève expansion [65]. Under the affine assumption of the random fields (4.2), the semi-weak formulation of the elliptic PDE (4.1) can be written as

$$A(u,v;y) = F(v;y) \quad \forall v \in H^1(D), \tag{4.3}$$

where the bilinear form A and the linear functional F can be expressed as

$$A(u,v;y) = \sum_{q=1}^{Q_a} \Theta_q^a(y) A_q(u,v) \text{ and } F(v;y) = \sum_{q=1}^{Q_f} \Theta_q^f(y) F_q(v), \tag{4.4}$$

with the definition $A_q(u,v) := (a_q \nabla u, \nabla v), 1 \le q \le Q_a$ and $F_q(v) := (f_q, v), 1 \le q \le Q_f$. Let us define a subspace $X \subset H_0^1(D)$ for the approximation of the PDE solution in physical space, for instance the high-fidelity finite element approximation space, spectral approximation space [54]. Then the reduced basis problem is formulated as: find $u_N \in X_N$ such that

$$A(u_N, v_N; y) = F(v_N; y) \quad \forall v_N \in X_N, \tag{4.5}$$

where the reduced basis space $X_N \subset X$ is constructed by span of the "snapshots", which are the solutions at N selected samples $y^1, \ldots, y^N$, i.e.

$$X_N = \text{span}\{u(y^n), 1 \le n \le N\}. \tag{4.6}$$

In order to guarantee algebraic stability in solving (4.5), we perform Gram-Schmidt orthogonalization procedure [61] on the snapshots $u(y^1), \ldots, u(y^N)$ and obtain a set of orthonormal bases $\zeta_1, \ldots, \zeta_N$ to form X_N. We expect the reduced basis space X_N to be a good approximation of the high-fidelity space X with dimension N as low as possible, so that the reduced basis problem (4.5) be very cheap to solve. To construct X_N, we develop in the next section a new adaptive greedy algorithm, which adopts a weighted a posteriori error bound [18] that will be briefly illustrated in section 4.3. In section 4.4, we present an offline-online decomposition that is particularly tailored for high-dimensional problems in order to efficiently evaluate the a posteriori error bound.

4.2 Adaptive greedy algorithm

In order to efficiently choose the N most representative samples while keeping the computational effort under control, we propose an adaptive greedy algorithm in combination with the construction of the dimension-adaptive hierarchical approximation in Algorithm 2. Given the reduced basis space X_N, the greedy algorithm seeks the next sample by maximizing the error between the reduced basis solution and the high-fidelity solution among all possible $y \in \Gamma$ [61, 52], i.e.

$$y^{N+1} = \underset{y \in \Gamma}{\operatorname{argmax}}\, \mathcal{E}_r(y), \text{ with } \mathcal{E}_r(y) := ||u(y) - u_N(y)||_X. \tag{4.7}$$

However, to solve problem (4.7) is computationally unfeasible since it is an infinity optimization problem, for which a full solve of a high-fidelity problem is needed to evaluate $u(y)$ at each $y \in \Gamma$. To cope with this unfeasibility, we propose an adaptive greedy algorithm based on the hierarchical approximation to render the infinity optimization problem as finite one in Algorithm 3, and use the a posteriori error bound as presented in the next section in order to avoid the full solve of the high-fidelity problem.

Algorithm 3 Adaptive greedy algorithm

1: **procedure** INITIALIZATION:
2: specify error tolerance ϵ_t, solve (4.3) at each $y \in \Theta^{\mathbf{1}}$ and construct $X_N = \text{span}\{u(y), y \in \Theta^{\mathbf{1}}\}$;
3: **end procedure**
4: **procedure** CONSTRUCTION:
5: at each step in line 9 of Algorithm 2, specify the set of nodes $\Theta^{rb}_{\triangle} = \Theta_{\triangle}$;
6: solve the reduced basis problem (4.5), compute $\mathcal{E}_r(y)$ and $s(y)$ at each $y \in \Theta^{rb}_{\triangle}$;
7: update $\Theta^{rb}_{\triangle}$ such that $\mathcal{E}_r(y) > \epsilon_t, \forall y \in \Theta^{rb}_{\triangle}$ (remove well approximated nodes);
8: **while** $\max_{y \in \Theta^{rb}_{\triangle}} \mathcal{E}_r(y) > \epsilon_t$ **do**
9: pick $y^{N+1} = \text{argmax}_{y \in \Theta^{rb}_{\triangle}} \mathcal{E}_r(y)$;
10: solve (4.3) at y^{N+1} and update $X_{N+1} = X_N \oplus \text{span}\{u(y^{N+1})\}$;
11: set $N = N + 1$ and repeat steps in line 6 - line 7 with new X_N;
12: **end while**
13: **end procedure**

We remark that the adaptive greedy algorithm 3 for the construction of reduced basis space explores all the nodes in the construction of the dimension-adaptive hierarchical approximation in Algorithm 2 and the outputs of interest s are evaluated based on the surrogate (reduced basis) solution with cheap solve of the reduced basis problem in contrast to the solution of the expensive high-fidelity problem. Moreover, error estimates of the surrogate outputs of interest can be obtained based on the reduced basis approximation error $\mathcal{E}_r$ controlled by the error tolerance ϵ_t.

4.3 Weighted a posteriori error bound

The reduced basis approximation error $\mathcal{E}_r$ used in the framework of the adaptive greedy algorithm plays a crucial role in constructing an efficient and accurate reduced basis space. In order to have a cheap, reliable and sharp evaluation of it, we adopt the residual based a posteriori error bound as

proposed in [61, 52]: for every $y \in \Gamma$, let $R(v;y) \in X'$ be the residual in the dual space of X, defined as

$$R(v;y) := F(v;y) - A(u_N(y), v; y) \quad \forall v \in X. \tag{4.8}$$

By Riesz representation theorem [23], we have a unique function $\hat{e}(y) \in X$ such that

$$(\hat{e}(y), v)_X = R(v;y) \quad \forall v \in X, \tag{4.9}$$

and $||\hat{e}(y)||_X = ||R(\cdot;y)||_{X'}$, where the X-norm is defined as $||v||_X = A(v, v; \bar{y})$ at some reference value $\bar{y} \in \Gamma$, e.g. the center of Γ. For the reduced basis error $e(y) := u(y) - u_N(y)$, we obtain the following equation as a result of (4.3), (4.5) and (4.8)

$$A(e(y), v; y) = R(v;y) \quad \forall v \in X. \tag{4.10}$$

By setting $v = e(y)$ and using Cauchy-Schwarz inequality, we have

$$\alpha(y)||e(y)||_X^2 \leq A(e(y), e(y); y) = R(e(y); y) \leq ||R(\cdot, y)||_{X'}||e(y)||_X = ||\hat{e}(y)||_X||e(y)||_X, \tag{4.11}$$

where $\alpha(y)$ is the coercivity constant of the bilinear form $A(e(y), e(y); y)$ at y, so that we can define the a posteriori error bound $\triangle_N^u(y)$ for the approximation error $||u(y) - u_N(y)||_X$ as

$$\triangle_N^u(y) := ||\hat{e}(y)||_X/\alpha(y), \tag{4.12}$$

yielding $||u(y) - u_N(y)||_X \leq \triangle_N^u(y)$ by (4.11). For the output in the compliant case, i.e. when $s(y) \equiv s(u(y); y) = F(u(y); y)$, we have the following error bound

$$|s(y) - s_N(y)| = |F(u(y); y) - F(u_N(y); y)| = A(e(y), e(y); y) \leq ||\hat{e}(y)||_X^2/\alpha(y) =: \triangle_N^s(y). \tag{4.13}$$

We remark that the error bounds (4.12) and (4.13) not only can be used as error indicator to construct the reduced basis space but also serve as certification of the reduced basis approximation, leading to the so called certified reduced basis method. In practice, the surrogate output of interest s_N evaluated based on the solution of the reduced basis problem (4.5) at any given node is a more accurate approximation of s than that obtained by the interpolation approach in Algorithm 2. Furthermore, the a posteriori error bound $\triangle_N$ is more reliable and accurate than the interpolation error indicator $\mathcal{E}_i$. Therefore, for pointwise evaluation in high dimensions, we employ the reduced basis method.

As for more general output where $s(y) \neq F(u(y); y)$, an adjoint problem of (4.3) can be employed to achieve faster convergence of the approximation error $|s - s_N|$, as will be illustrated later. In order to take arbitrary probability measure into account for efficient numerical integration, we employ the weighted a posteriori error bound developed in [18] as

$$\triangle_N^{\rho,u}(y) = \sqrt{\rho(y)}\triangle_N^u(y), \text{ or } \triangle_N^{\rho;s}(y) = \rho(y)\triangle_N^s(y), \tag{4.14}$$

which put small weight on the samples with small probability density, thus generating relatively less bases while achieving the same accuracy of total integration for UQ problems with non-uniform distributed random variables. More details about convergence analysis and illustrative examples are provided in [18]. In summary, computing the error indicators (4.14) requires the evaluation of the coercivity constant $\alpha(y)$ and the value $||\hat{e}(y)||_X$ at given $y \in \Gamma$. For the former, we may use the successive constraint linear optimization method [38] to compute a pointwise lower bound $\alpha_{LB}(y) \leq \alpha(y)$, or simply use a uniform lower bound $\alpha_{LB} \leq \alpha(y)$ that holds for all $y \in \Gamma$, in order to alleviate the computational effort, provided that the coercivity constants at different samples $y \in \Gamma$ are not very different. For the latter, we adopt an offline-online computational decomposition that we will illustrate in the next section.

4.4 Offline-online decomposition

In the construction of the reduced basis space with a small number of bases and the evaluation of a large number of outputs of interest, we efficiently split the computational work for the former at an offline stage from that for the latter at an online stage by taking advantage of the affine structure (4.2). To start, we expand the reduced basis solution $u_N(y)$ on the reduced bases as

$$u_N(y) = \sum_{m=1}^{N} u_{Nm}(y)\zeta_m, \tag{4.15}$$

which is substituted in the Galerkin projection problem (4.5) as: find $u_{Nm}(y), 1 \le m \le N$ such that

$$\sum_{m=1}^{N}\sum_{q=1}^{Q_a} \Theta_q^a(y) A_q(\zeta_m, \zeta_n) u_{Nm}(y) = \sum_{q=1}^{Q_f} \Theta_q^f(y) F_q(\zeta_n), \quad 1 \le n \le N. \tag{4.16}$$

Here, the matrix $A_q(\zeta_m, \zeta_n), 1 \le q \le Q_a, 1 \le m, n \le N$ and the vector $F_q(\zeta_n), 1 \le q \le Q_f, 1 \le n \le N$ can be pre-computed and stored in the offline stage. In the online stage, we only need to assemble and solve the resulting $N \times N$ stiffness system of (4.16) with much less computational effort compared to solving the original high-fidelity stiffness system. The approximate compliant output $s_N(y)$ is thus evaluated by $N \times Q_f$ operations as

$$s_N(y) = F(u_N(y); y) = \sum_{n=1}^{N} \left(\sum_{q=1}^{Q_f} \Theta_q^f(y) F_q(\zeta_n) \right) u_{Nn}(y). \tag{4.17}$$

As for the evaluation of $||\hat{e}(y)||_X^2$ in (4.13), we first expand the residual (4.8) as

$$R(v; y) = F(v; y) - A(u_N, v; y) = \sum_{q=1}^{Q_f} \Theta_q^f(y) F_q(v) - \sum_{n=1}^{N} \left(\sum_{q=1}^{Q_a} \Theta_q^a(y) A_q(\zeta_n, v) \right) u_{Nn}(y). \tag{4.18}$$

Then, let $\mathcal{C}_q \in X$ such that $(\mathcal{C}_q, v)_X = F_q(v), \forall v \in X, 1 \le q \le Q_f$ and $\mathcal{L}_q^n \in X$ such that $(\mathcal{L}_q^n, v)_X = -A_q(\zeta_n, v), \forall v \in X, 1 \le n \le N, 0 \le q \le Q_a$, which can be regarded as the Riesz representatives of F_q and A_q^k (defined as $A_q^k(v) = -A_k(\zeta_n, v), \forall v \in X$) in X. By recalling (4.9) we have

$$\hat{e}(y) = \sum_{q=1}^{Q_f} \Theta_q^f(y) \mathcal{C}_q - \sum_{n=1}^{N} \left(\sum_{q=1}^{Q_a} \Theta_q^a(y) \mathcal{L}_q^n \right) u_{Nn}(y), \tag{4.19}$$

so that

$$\begin{aligned} ||\hat{e}(y)||_X^2 = & \sum_{q}^{Q_f}\sum_{q'}^{Q_f} \Theta_q^f(y)\Theta_{q'}^f(y)(\mathcal{C}_q, \mathcal{C}_{q'})_X \\ & + 2\sum_{n=1}^{N}\sum_{q=1}^{Q_f}\sum_{q'=1}^{Q_a} \Theta_q^f(y)\Theta_{q'}^a(y)(\mathcal{C}_q, \mathcal{L}_{q'}^n)_X u_{Nn}(y) \\ & + \sum_{n=1}^{N}\sum_{n'=1}^{N}\sum_{q=1}^{Q_a}\sum_{q'=1}^{Q_a} \Theta_q^a(y)\Theta_{q'}^a(y) u_{Nn}(y)(\mathcal{L}_q^n, \mathcal{L}_{q'}^{n'})_X u_{Nn'}(y). \end{aligned} \tag{4.20}$$

Therefore, we can pre-compute and store $(\mathcal{C}_q, \mathcal{C}_{q'})_X, 1 \le q, q' \le Q_f, (\mathcal{C}_q, \mathcal{L}_{q'}^n)_X, 1 \le n \le N, 1 \le 1 \le Q_f, 1 \le q' \le Q_a, (\mathcal{L}_q^n, \mathcal{L}_{q'}^{n'})_X, 1 \le n, n' \le N, 1 \le q, q' \le Q_a$ in the offline stage, and evaluate $||\hat{e}(y)||_X$ in the online stage by assembling (4.20) with $O(Q_f^2 + NQ_fQ_a + N^2Q_a^2)$ operations.

Note that when the number of terms Q_f and Q_a become large, the full online evaluation of (4.20)

will be expensive. Let us make two observations in order to further reduce the online evaluation cost: the first is that often $\Theta_q^a(y) = y_q$ with Q^a representing the dimension of a high-dimensional probability space for UQ problems; the other is that the nodes inside one set $\Theta_\triangle$ or from neighbor sets are only different from each other in limited dimensions, e.g. the node $(1, 0.5, 0.5, \ldots, 0.5)$ is a neighbor of the node $(0, 0.5, 0.5, \ldots, 0.5)$, which are only different in the first dimension. Based on these two observations, we may identify the different terms in (4.20) from one node to the next in the adaptively constructed grid and only subtract these terms from $||\hat{e}(y)||_X^2$ at the previous node and add the corresponding new terms to it at the current node, resulting in $O(Q_f + NQ_a)$ operations in average for each evaluation. We remark that this computational reduction is still valid whenever there are only a few terms among $\Theta_q^a(y), 1 \leq q \leq Q_a$ different from one node to its neighbors.

4.5 Remarks on extension to more general PDE models

We presented the adaptive and weighted reduced basis method based on a coercive, steady and linear elliptic equation with affine input and compliant output. However, the method is not constrained by these elementary properties. In fact, it has been developed and extended to deal with many different PDE models, [62, 60, 35, 56, 4, 31, 17, 21], and applied in a variety of physical and engineering fields, [55, 19, 56, 40, 59, 12, 14]. We provide the following remarks for extensions with some associated references.

First of all, the coercivity property (4.11) is used in computing a lower bound for the evaluation of a posteriori error bound (4.12). When the problem fails to be coercive, for instance in Stokes equations, where only an "inf-sup" compatibility condition is satisfied, we can replace the coercivity constant α in (4.11) by an *inf-sup constant* and arrive at the same reliable and accurate a posteriori error bound [62, 60]. Moreover, we may even sacrifice the reliability of the error bound if the "inf-sup" condition is not satisfied either and only use the residual as an error indicator.

Secondly, for unsteady problems, the reduced bases should be explored not only at different samples but also at different time steps. In order to efficiently extract the most representative bases, we may employ proper orthogonal decomposition (POD) to project the solutions at different time steps into a small number of bases and use a greedy algorithm to choose the samples, leading to a *POD-greedy algorithm* [35, 56].

Thirdly, in order to deal with nonlinear problems, different approaches can be adopted. Taking Navier-Stokes equations for example, where the nonlinearity is quadratic on the state variable, we may employ Newton iteration to solve the reduced basis system as done for solving the high fidelity system [55]. Another approach is to use the *empirical interpolation for operators* [31, 21] in decomposing the nonlinear operators into linear combination of a series of linear operators.

Fourthly, when the random inputs are not given in affine structure, e.g. log-normal random field, we may reconstruct the nonaffine random field as random field with finite affine terms by *empirical interpolation method* [4, 31, 17]. This reconstruction is very efficient (resulting in a limited number of affine terms) for smooth functions and functions that enjoy the compressibility property, i.e. a function is compressible from a high-dimensional space to a low-dimensional space without losing too much accuracy.

Finally, for non-compliant problems where the output of interest is different from the right hand side of the equations, the error convergence for the approximation of the output depends only linearly on the error convergence for the approximation of the solution. Moreover, the norm of the functional for the output may not be easy to evaluate. In this case, we employ a *primal-dual approach*, where a dual problem is formulated by setting the output as a right hand side of the dual equations and the output is evaluated with contribution from both the primal and dual problem [61, 52]. The advantage is that it avoids the computation of the functional norm and achieves quadratic error convergence. We will illustrate this approach by one numerical example in section 5.6.

5 Numerical experiments

This section is devoted to demonstrate the efficiency and accuracy of the adaptive and reduced computational framework and compare it to other methods (anisotropic sparse grid, ANOVA as introduced

in section 3.4) for high dimension uncertainty quantification problems. We illustrate the computational performance of the proposed Algorithm 2 with verification in two dimensions, and compare it to the algorithm without verification and anisotropic sparse grid scheme (3.26) in section 5.1. In section 5.2 we illustrate why the ANOVA approach does not work well for functions with strong interaction and arbitrary probability measure, whereas this case can be efficiently dealt with by our proposed approach. In section 5.4, we show how the sparsity in high dimensions (from $O(10)$ to $O(1000)$), including different interaction and importance of different dimensions, can be efficiently and accurately captured by the proposed method. The last two sections 5.5 and 5.6 deal with heat diffusion and groundwater flow problems and demonstrate how the adaptive and weighted reduced basis method can be effectively applied to reduce the computational effort.

5.1 Hierarchical construction with verification

In this experiment, we compare the dimension-adaptive hierarchical interpolation Algorithm 2 with the same algorithm without the procedure of verification. The two dimensional function $s : [0, 1]^2 \to \mathbb{R}$ is given by

$$s(y) = \cos(2\pi(y_1 - 0.3)) \cos(2\pi(y_2 - 0.5)). \tag{5.1}$$

We run the interpolation Algorithm 2 in six different cases. The first three cases include hierarchical construction without verification based on piecewise linear polynomials with equidistant nodes and the weight in (3.25) are set as $w = 1, 0.5, 0$, corresponding to the purely dimension-adaptive grid construction, balanced construction and conservative sparse grid construction, respectively. The fourth case is specified with the same configuration as the first three except that the verification procedure is incorporated. The last two cases use Lagrange polynomials based on Clenshaw–Curtis nodes with verification and the weight $w = 1$ and $w = 0$, respectively. We set the maximum number of nodes adaptively as one larger than the number of nodes in the current grid, with the upper bound $M = 10^4$, and specify the interpolation error tolerance as $\varepsilon_t = 10^{-15}$. We compute the interpolation error as $\max_{y\in\Xi_{test}} |s(y) - \mathcal{S}_g s(y)|$ with the set of testing nodes given by $\Xi_{test} := \{y_1, y_2 = n/2^8, n = 0, \ldots, 2^8\}$, a fine regular grid with step size $1/2^8$. The final index sets S_m for the six different cases are plotted in Figure 5.1, where the active indices are marked with boxes (blue and red) and the index to be processed in the next step is marked with red box. Figure 5.2 reports the interpolation errors for all the six cases.

From the first figure (left-top of Figure 5.1), we can see that the enrichment of active indices has stagnated along y_2 by the purely dimension-adaptive scheme, resulting in large interpolation error (see left of Figure 5.2) since the function is not sufficiently well approximated in the second dimension. The balancing scheme with $w = 0.5$ (see middle-top of Figure 5.1) is able to construct fine grid in the second dimension but fails to capture the interaction of the two dimensions (due to stagnation), and thus still leads to large interpolation error as shown in Figure 5.2 (left). The Smolyak sparse grid construction introduced in section 3.2 does not run into the stagnation problem and achieves small interpolation error in this example (see right-top of Figure 5.1), but it can identify neither the important dimension nor the interaction. This drawback can be observed more clearly by comparison of the grid construction in the last two cases, where a full tensor grid is constructed by the adaptive scheme (see middle-bottom of Figure 5.1) and the sparse scheme that produces many more useless nodes in each single dimension (see right-bottom of Figure 5.1). Note that the last two cases result in higher approximation accuracy (see Figure 5.2) than the others because the globally supported Lagrange polynomial basis is more suitable to approximate smooth functions. By using the same locally supported piecewise linear basis as in the first three cases but incorporating the verification procedure, we can get rid of the stagnation problem and adaptively construct the grid with automatic identification of the importance and interaction of different dimensions, as shown in Figure 5.1 (left-bottom). From this experiment (two dimensional case for the sake of the illustration), we can see that the verification procedure works efficiently to get rid of the stagnation problem, which is to blame as one drawback of the dimension-adaptive hierarchical construction approach. We remark that the balancing scheme in (3.25) can not effectively avoid stagnation. Moreover, it is not computationally convenient to use since the weight parameter w is not known a priori and it depends on different problems under consideration.

In the second example, we test the efficiency of the verified dimension-adaptive algorithm for

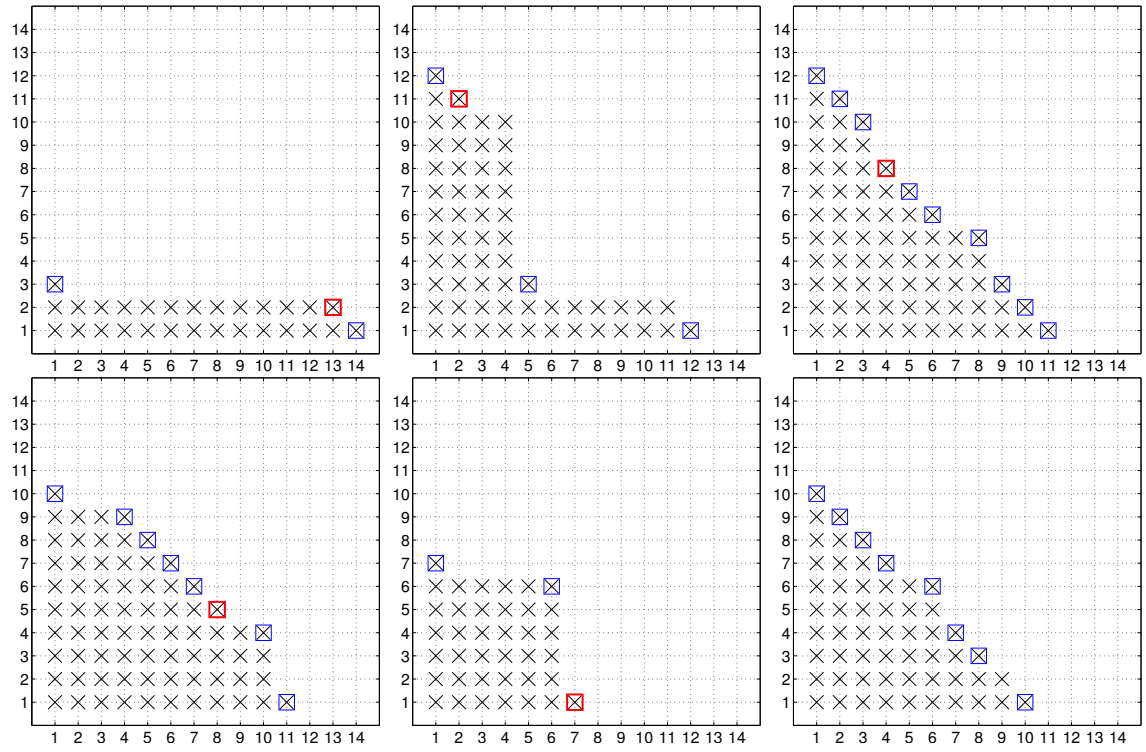

Figure 5.1: Illustration of dimension-adaptive hierarchical construction of the generalized sparse grid in different cases; top row: piecewise interpolation without verification with weight $w = 1$ (left), $w = 0.5$ (middle), and $w = 0$ (right); bottom row: piecewise interpolation with verification and weight $w = 1$ (left), Lagrange interpolation with verification and weight $w = 1$ (middle), and $w = 0$ (right).

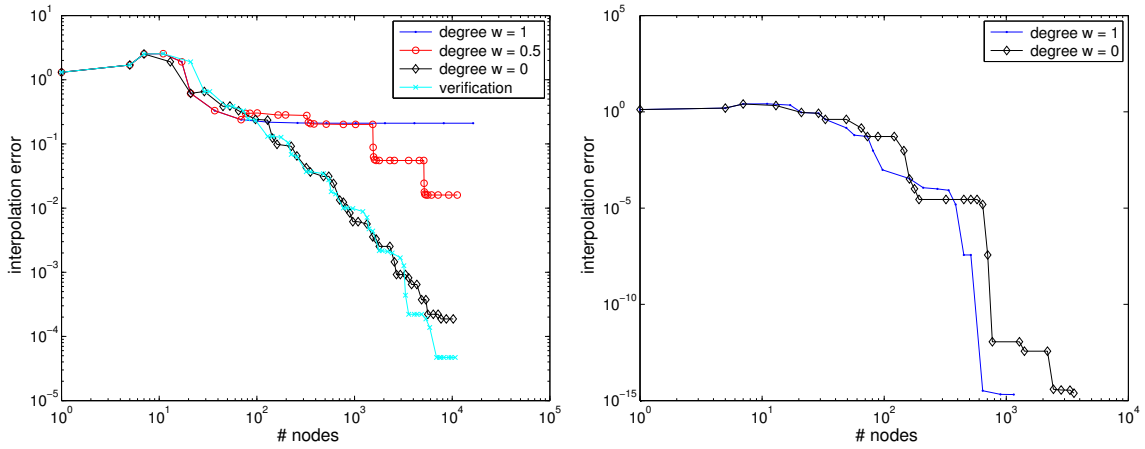

Figure 5.2: Interpolation error corresponding to the grid construction in Figure 5.1; left: picecwise interpolation in the first four cases; right: Lagrange interpolation in the last two cases.

interpolation of these anisotropic functions

$$s_1(y) = \exp(y_1/5)+\exp(5y_2), \quad s_2(y) = \exp(y_1y_2), \quad s_3(y) = \exp(y_1/5)+\exp(5y_2)+\exp(y_1y_2). \quad (5.2)$$

We run the interpolation Algorithm 2 with the interpolation error tolerance set as $\varepsilon_t = 10^{-15}$. The constructed indices are displayed in Figure 5.3, from which we can see that the verified dimension-

adaptive algorithm efficiently and accurately captured the interaction and importance of different variables of the test functions. The first one has no interaction term and y_2 plays a more important role (in terms of function value) than y_1. The second one features strong interaction and equal importance of the two dimensions. The last one has strong interaction and more important dimension y_2 than y_1. We remark that these properties can not be captured by the anisotropic sparse grid construction with weighted index set (3.26) as introduced in [46]. As a matter of fact, such approach either deteriorates efficiency because many useless indices are included or loses accuracy because the necessary indices (for strong interaction term) can not be captured, especially in high dimensions.

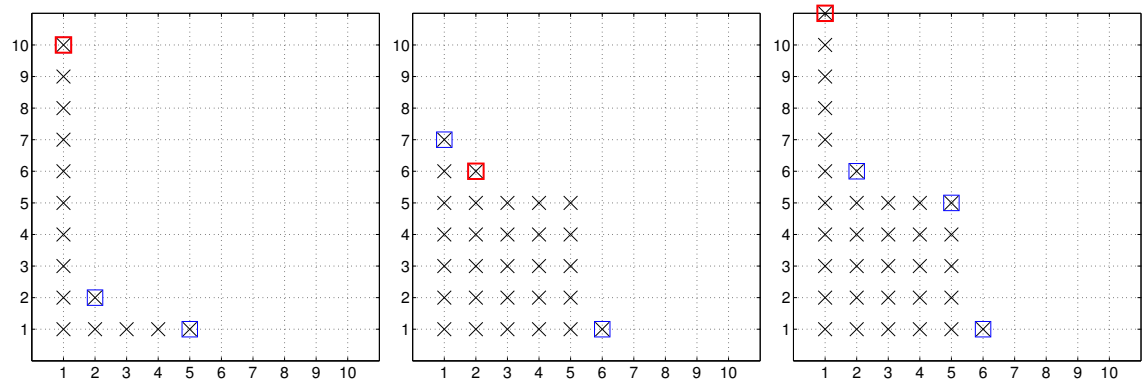

Figure 5.3: Illustration of dimension-adaptive hierarchical construction of generalized sparse grid for anisotropic interpolation; indices for s_1 (left), s_2 (middle), s_3 (right) with tolerance $\varepsilon_t = 10^{-15}$.

5.2 Sobol functions featuring strong interaction

In this numerical experiment, we study the functions with separated variables proposed by Sobol [68] to test the accuracy and efficiency of the hierarchical approximation in the extreme case - building minimal full tensor product grid, with comparison to the approximation based on anchored ANOVA (cut-HDMR) [26, 42]. The functions are defined as

$$s_1(y) = \prod_{k=1}^{K} \frac{|4y_k - 2| + p_k}{1 + p_k} \text{ and } s_2(y) = \prod_{k=1}^{K} \frac{1 + 3p_k y_k^2}{1 + p_k}, \tag{5.3}$$

where $y_k \in [0,1], 1 \le k \le K$ and the parameter $p_k, 1 \le k \le K$, is nonnegative for the first function and positive for the second one. Both functions have separated variables, meaning that the total integral (with value 1) can be computed by the product of individual integrals evaluated separately, but all of them are strongly interacting for pointwise evaluation of the function value. Since the first function has singularities ("peaks") at $y_k = 0.5, 1 \le k \le K$ and the second function is smooth, we use piecewise polynomial basis for the first function and global Lagrange polynomial basis for the second one. First of all, let us take a simple low-dimensional function s_2 with $K = 3$ and $p_k = 1, 1 \le k \le K$ and consider the anchored ANOVA approximation with several different anchor points $\bar{y}_k, 1 \le k \le K$ and expansion orders. Let $\bar{s}_i, 0 \le i \le K$ denote the approximated integral with expansion up to i dimensions (see the expansion formula (3.28)). When $i = 0$, the approximated integral is taken as the function value at the anchor point. The approximated integrals for different additive functions in the expansion are computed by tensor product Clenshaw–Curtis quadrature formula with 3 abscissas in each dimension. The results at different settings are reported in Table 5.1, from which we can observe that the approximation results are far from each other at different anchor points before the full expansion with $i = K$ is used. Moreover, the averaged approximations in the last column do not lead to a more accurate approximation as proposed in [34]. The approximations of the integral converge to the exact value with growing expansion order and reach the exact value only when the full expansion with $2^K = 8$ terms has been incorporated in all cases. These observations confirm

the drawbacks of the anchored ANOVA approximation as pointed out in section 3.4. Similar results can be shown also for the first function s_1 and for higher dimensional integration problems by this approach. In fact, there is no gain in this case but more cost by the anchored ANOVA approximation since not only the last term has to be evaluated in all the K dimensions but also the other $2^K - 1$ terms of the expansion (3.28).

$\bar{y}_k$	0.0000	0.1667	0.3333	0.5000	0.6667	0.8333	1.0000	average
$\bar{s}_0$	0.1250	0.1589	0.2963	0.6699	1.5880	3.6641	8.0000	2.0717
$\bar{s}_1$	0.5000	0.5624	0.7407	0.9570	0.9074	-0.1981	-4.0000	-0.0758
$\bar{s}_2$	0.8750	0.9037	0.9630	0.9980	1.0046	1.1589	2.0000	1.1290
$\bar{s}_3$	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000

Table 5.1: Approximated values of the integral of the function s_2 by anchored ANOVA expansion (3.28) with different anchor points (in row) and expansion orders (in column).

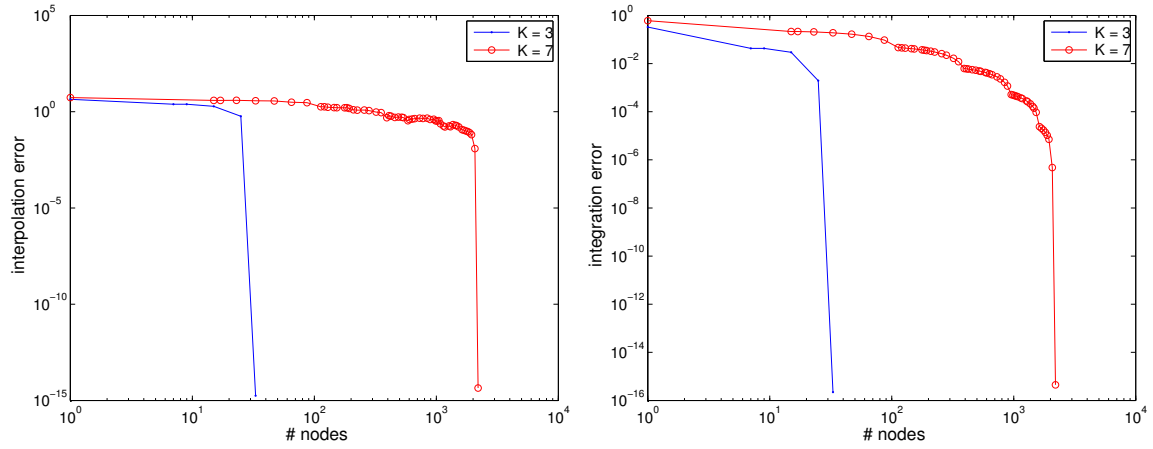

Figure 5.4: Interpolation error (left) and integration error (right) of dimension-adaptive hierarchical approximation of the smooth function s_2 with the dimension $K = 3$ and $K = 7$.

The interpolation and integration errors for the dimension-adaptive hierarchical approximation of the smooth function s_2 are displayed in Figure 5.4, where the interpolation error is defined by $\max_{y \in \Xi_{test}} |s(y) - \mathcal{S}_g s(y)|$ with the testing set Ξ_{test} consisting of 100 randomly selected samples. The decay of both interpolation and integration errors is very slow at the beginning, and fall to about the machine precision when the minimal full tensor product grid with 3 nodes in each dimension has been constructed, requiring in total $3^3 = 27$ and $3^7 = 2187$ nodes, respectively. This decay confirms again the necessity to use all the expansion terms by the anchored ANOVA approximation in order to have accurate integration. The dimension-adaptive hierarchical algorithm successfully detects the full tensor product grid structure and construct it automatically with the ultimate number of nodes 33 and 2201, slightly bigger than those of the full tensor product grid due to the verification procedure. As for the approximation of the singular function s_1, we reduce the effect of the variation of y_k by setting a large parameter $p_k = 100, 1 \leq k \leq K$, which leads to the results in dimensions $K = 4$ and $K = 8$ in Figure 5.5. Similar convergence behaviour can be observed for the singular function as that for the smooth function, in particular 89 and 6577 nodes are constructed close to the minimal number of full tensor product grid $3^4 = 81$ and $3^8 = 6561$. Note that in this case, the approximation errors decay more uniformly due to the reduced variation.

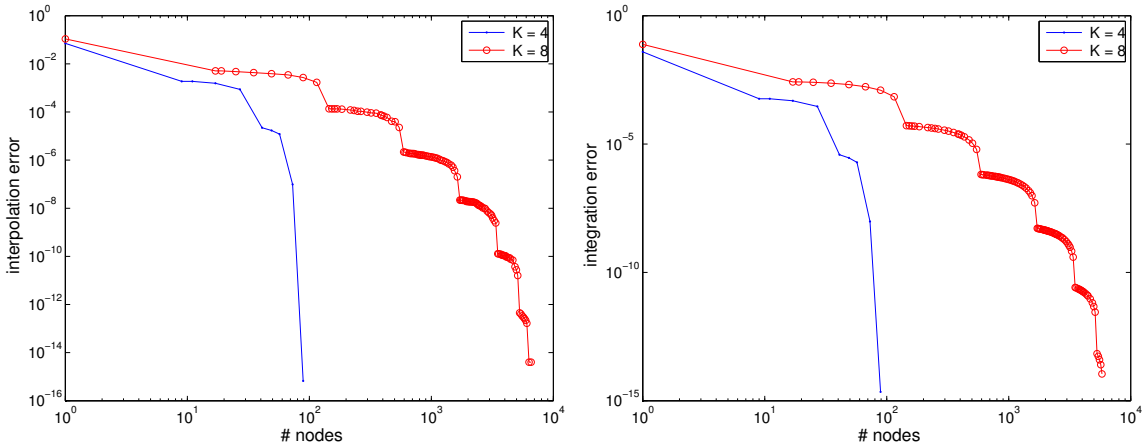

Figure 5.5: Interpolation error (left) and integration error (right) of dimension-adaptive hierarchical approximation the singular function s_1 with dimension $K = 4$ and $K = 8$.

5.3 Approximation with arbitrary probability measure

By this experiment, we study the dimension-adaptive hierarchical approximation with arbitrary probability measure in order to demonstrate the efficiency of using the interpolation and integration error indicators (3.22) and (3.8) for interpolation and integration problems, respectively, and illustrate why the variance-based ANOVA (or HDMR) expansion is not suitable for interpolation problems. We use the following exponential function

$$s(y) = \exp\left(-\sum_{k=1}^{K} c_k(y_k - 0.5)\right), \tag{5.4}$$

and set $c_k = 1, 1 \leq k \leq K$ with dimension $K = 5$. The random variables are set to obey beta distribution as $y_k \sim Beta(\beta k, \beta k), 1 \leq k \leq K$, being $\beta \in \mathbb{R}_+$ a scaling parameter. The probability density function (PDF) with different parameters is displayed in Figure 5.6, from which we can see that as the parameter becomes bigger, the more concentrated the PDF becomes and the smaller the variance is. Therefore, the importance of different dimensions becomes different as influenced by the given probability measure instead of the parameter c_k.

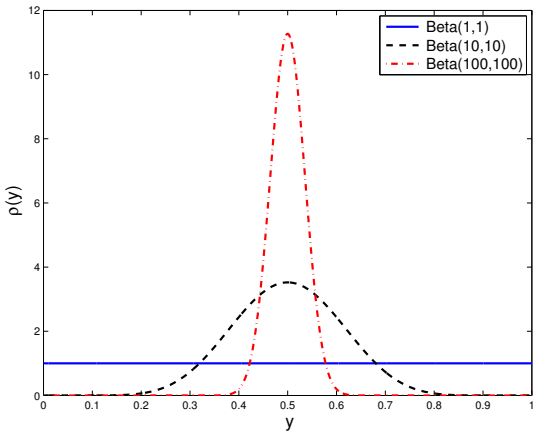

Figure 5.6: Probability density function of beta distributed random variable with different parameters.

We run the dimension-adaptive hierarchical approximation Algorithm 2 to compute both the

interpolation and the integration of the given function with different error indicators. We employ the nested Kronrod-Patterson quadrature nodes [53] associated with the beta measure at different parameter β. The interpolation error is computed as the maximum at 100 randomly selected samples and the integration error is computed by taking the approximation of the integral in the final step as the "exact" value. The error convergence of the approximation for interpolation and integration with different error indicators is shown for $\beta = 1, 5, 10, 20$ in Figure 5.7. From the left of Figure 5.7, we can see that the interpolation errors obtained with interpolation error indicator converge faster than those obtained with integration error indicators at different values of β. On the other hand, the convergence of the integration errors shown on the right of Figure 5.7 highlights that the integration error indicator leads to evidently more accurate approximation of the integral than the interpolation error indicator for the cases $\beta = 5, 10, 20$. These observations confirm that the integration error indicator, closely related by the underlying probability measure to the variance-based ANOVA approximation, works efficiently for integration but may give rise to large errors for interpolation.

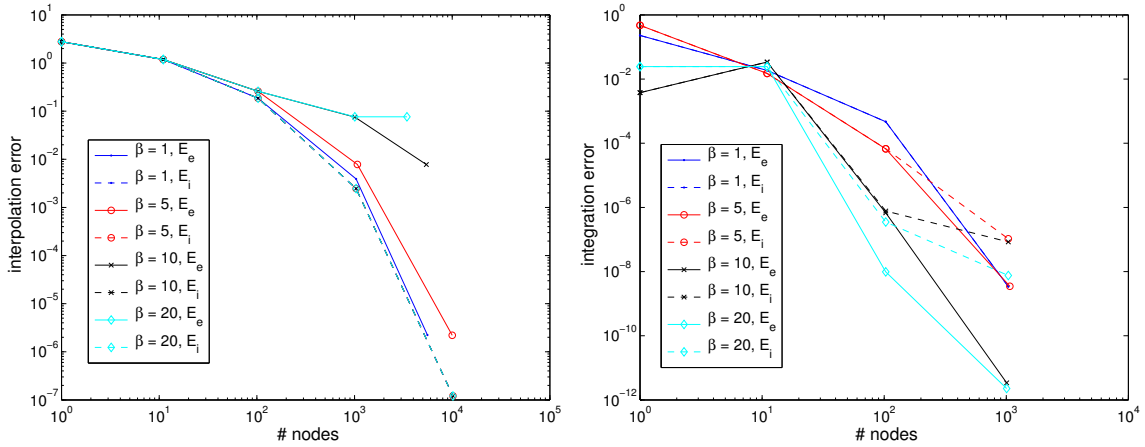

Figure 5.7: Interpolation error (left) and integration error (right) with different scaling parameter $\beta = 1, 5, 10, 20$ (with different markers) and error indicators, $\mathcal{E}_i$ in dashed line and $\mathcal{E}_e$ in solid line.

5.4 High-dimensional functions featuring sparsity

In this numerical experiment, we test the performance of the dimension-adaptive hierarchical approximation of high-dimensional functions featuring sparsity, i.e. low interaction or distinct importance of different dimensions. The first function has low interaction property, given by

$$s(y) = \sum_{k=1}^{K} y_k^2 - \sum_{k=1}^{K-1} y_k y_{k+1}, \tag{5.5}$$

which is a polynomial of total degree 2 and interaction level 2 (in the sense of ANOVA expansion (3.28)). We set the dimension as $K = 10$ and 100, and run the dimension-adaptive hierarchical approximation algorithm with the interpolation error indicator (3.22) for both the interpolation and integration. The interpolation error is computed at 100 randomly selected samples, and the integration error is measured with respect to the value of the exact integral $K/3 - (K-1)/4$. Since the function is smooth, we use Lagrange basis with Clenshaw–Curtis nodes. The results of the error indicator, the interpolation and integration errors, as well as the number of nodes are reported in Table 5.2, from which we can see that the second level isotropic sparse grid is sufficient to evaluate the integral accurately up to machine precision (with rounding error) in all the three cases, while for accurate interpolation the third level of sparse grid is needed and sufficient due to the interaction in the second term of s. The dimension-adaptive hierarchical construction algorithm is able to detect the isotropic structure of the sparsity and build automatically the isotropic sparse grid as can be seen from the

comparison of the number of nodes in Table 5.2, where a small number of extra nodes are used for checking stopping criterion. We remark that in order to detect the full interaction relation of s by ANOVA expansion, the total number of terms to be explored is $1 + K + (K-1)(K-2)/2$, which results in the same number of nodes by sparse grid or two dimensional tensor product grid (3 nodes in each dimension) for each term of the second level.

K	level	# nodes	error indicator $\mathcal{E}_i$	interpolation error	integration error
10	1	1 (1)	∞	1.9499	0.8333
	2	21 (21)	0.5000	0.6306	6.661e-16
	3	233 (221)	3.8858e-16	1.332e-15	6.661e-16
100	1	1 (1)	∞	11.5087	8.3333
	2	201 (201)	0.5000	1.9956	1.243e-14
	3	20213 (20201)	3.5527e-15	4.796e-15	1.066e-14

Table 5.2: Interpolation and Integration errors for dimension-adaptive hierarchical approximation of the low interacting function s_1; the number of nodes in $(\cdot)$ corresponds to an isotropic sparse grid.

We use (5.4) as the second test function, which features the sparsity due to distinct importance even with strong interaction of different dimensions. Here the parameter $c_k \in \mathbb{R}_+$ determines the importance of the dimension $k = 1, \ldots, K$; $y_k \in [0,1], 1 \le k \le K$ are independent and uniformly distributed random variables. In the first example, we set $c_k = \alpha^{-k+1}, 1 \le k \le K$, with the scaling parameter $\alpha = 1.1$ and consider the dimension $K = 2^n, 3 \le n \le 6$. Clenshaw–Curtis quadrature is employed for the computation of the integral, where the "exact" value is taken as the approximation at the last step. We set the maximal number of nodes as $M = 10^m, 1 \le m \le 5$. The interpolation and integration error convergence is depicted in Figure 5.8 and the level of interpolation (note that we plot $i_k - 1$ in y axis due to implementation convenience) in the 64 dimensional case is reported in Figure 5.9. From these two figures we can conclude that only the first few dimensions dominate all the other dimensions and the dimension-adaptive hierarchical approximation Algorithm 2 successfully constructed the grid according to the importance of different dimensions. The convergence rate of the integration error for the 64 dimensional problem is around 1, which is faster than that of the Monte Carlo method (rate $= 1/2$) or quasi Monte Carlo method (rate $\in (1/2, 1)$) [20].

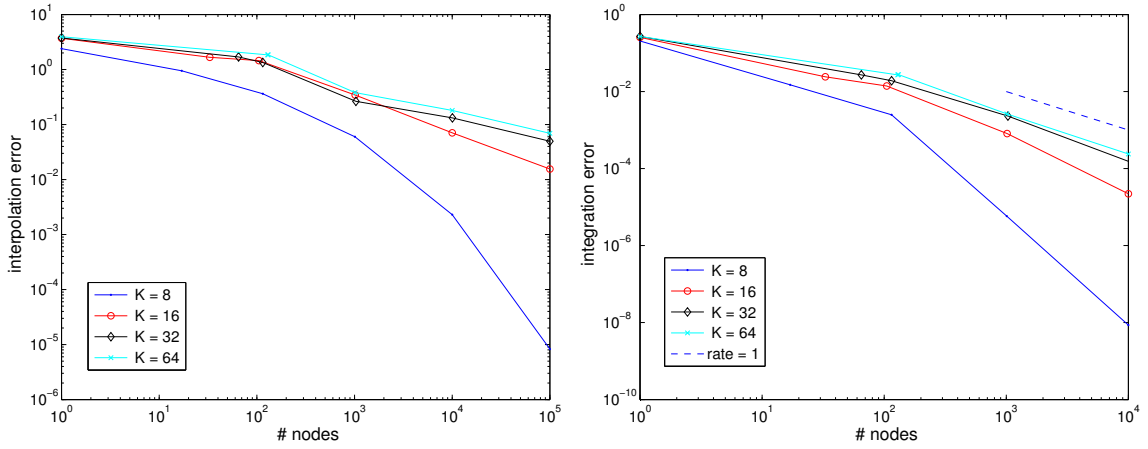

Figure 5.8: Convergence of interpolation error and integration error with dimension $K = 8, 16, 32, 64$.

In the second example, we test the dimension $K = 100, 400, 900, 1600$ and set the parameter $c_k, 1 \le k \le K$ as follows: we randomly select $\sqrt{K}$ dimensions and set c_k in these dimensions as $10^{-y_k^0}$, where $y_k^0 \in [0,1]$ is a sample drawn from uniform distribution, and in the other dimensions we set $c_k = 10^{-y_k^0 - 6}$. Therefore, the dimensions are divided into two scales. In each scale the importance of different dimensions is determined by a random variable $10^{-y_k^0}$. In another word, the important

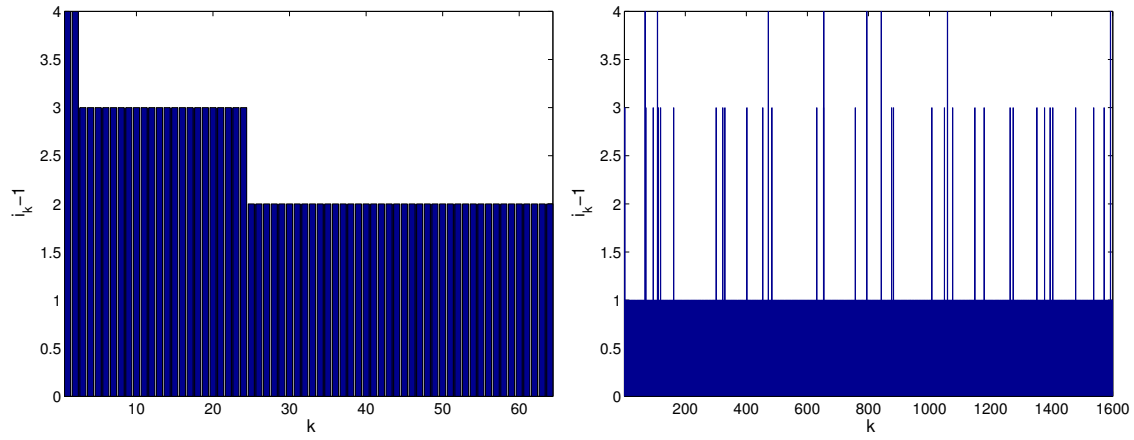

Figure 5.9: Grid level constructed by the dimension-adaptive hierarchical aproximation algorithm 2.

dimensions randomly distributes from 1 to K with total effective number of dimensions around $\sqrt{K}$. The convergence results for the interpolation and integration error is shown in Figure 5.10, from which we can see that the dimension-adaptive hierarchical approximation works efficiently for high-dimensional problems, with the integration error converging faster than Monte Carlo method with the total dimension as high as 1600. The right of Figure 5.9 demonstrates that both the scales (between level $i_k = 1+1$ and levels $i_k = 1+3, 1+4$) and the importance in each scale (between level $i_k = 1+3$ and $i_k = 1+4$) of different dimensions are captured effectively by the dimension-adaptive algorithm 2. We remark that the examples in high-dimensional space feature distinct importance of different dimensions. In the case of equal importance of different dimensions in high-dimensional problems, the classical Monte Carlo method would achieve better computational performance.

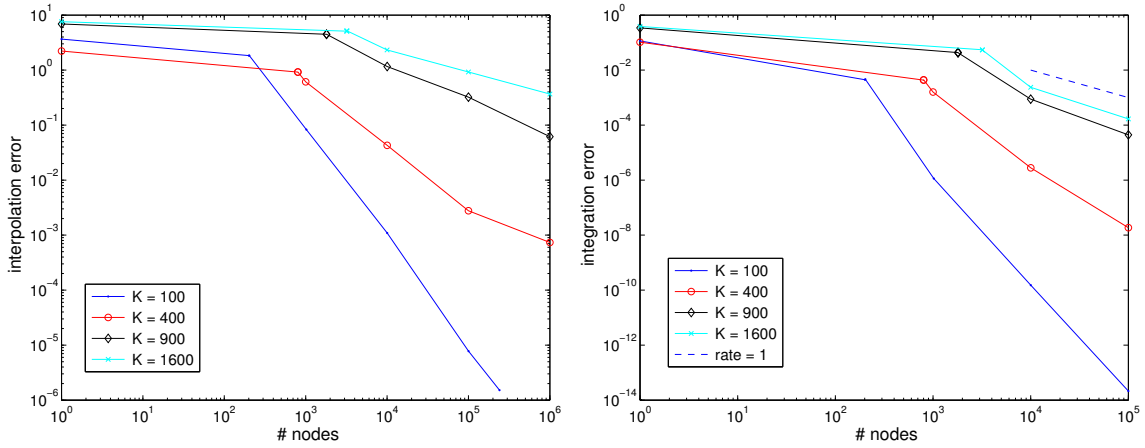

Figure 5.10: Convergence of interpolation error and integration error with dimension $K = 100, 400, 900, 1600$.

5.5 Heat diffusion in thermal blocks

In this example, we study a heat diffusion problem (4.1) in thermal blocks with the thermal conductivity modeled by random variables. The problem is defined in the physical domain $D = (0,1)^2$ discretized with 101^2 nodes, which can be equally divided into K ($K = n^2, n \in \mathbb{N}_+$) blocks $D_k, 1 \le k \le K$. The thermal conductivity of each block is a random variable. In the first test, we demonstrate the effi-

ciency of the weighted a posteriori error bound (4.14) in the case of arbitrary probability measure for integration problem. We consider the random coefficient a in (4.1) as

$$a(x,y) = \sum_{k=1}^{K} \chi_{D_k}(x) 10^{(y_k - 0.5)}, \tag{5.6}$$

where χ_{D_k} is a characteristic function supported on the block D_k and $y_k \in [0,1], 1 \le k \le K$ with $K = 9$, are independent random variables obeying beta distribution $Beta(\beta,\beta)$ with $\beta = 5$, which feature almost equal importance in each of the 9 dimensions. A deterministic force term is considered as $f = 1$. We run the adaptive greedy Algorithm 3 with tolerance $\epsilon_t = 10^{-11}$ to construct the reduced basis space based on the hierarchical construction of the generalized sparse grid by Algorithm 2. For the construction of the generalized sparse grid, the integration error indicator (3.23) is used with the total number of nodes specified as $10^n, 0 \le n \le 4$ and the nested Kronrod-Patterson quadrature nodes are employed corresponding to the beta measure with different parameter β. The quantity of interest is the average temperature over the whole domain $\int_D u dx$, which is compliant as in (4.13). We apply both the a posteriori error bound (4.12) and the weighted a posteriori error bound (4.14) to construct the reduced basis space, resulting in 211 and 118 bases, respectively.

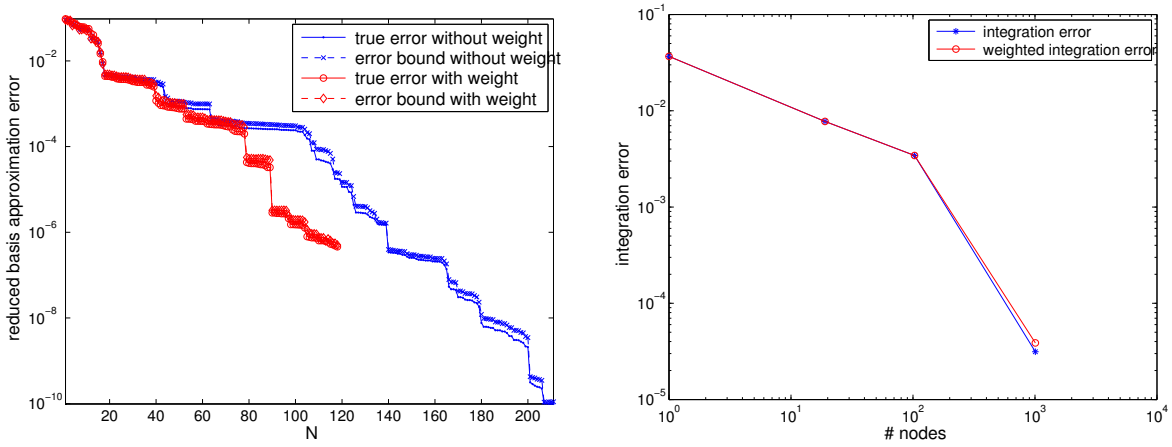

Figure 5.11: Left: true error and error bound of reduced basis approximation constructed by a posteriori error bound without (4.12) and with weight (4.14); right: (weighted) integration error.

The reduced basis approximation error (in the worse scenario case) tested with 100 randomly samples and the integration error (computed with the integral at 10^4 nodes as the reference value) of the two different cases are depicted in Figure 5.11. From the right of this figure we can see that the reduced basis approximation with the weighted a posteriori error bound (4.14) achieves almost the same accuracy for integration as that without the weight (4.12), even using much less bases (118 compared to 211). As for the pointwise approximation, the weighted scheme results in faster convergence of the reduced basis approximation error than that without the weighted scheme, though does not guarantee the same small error at the end because it makes use of much less reduced bases, see in the left of Figure 5.11. Moreover, from the comparison of the true error and error bound plotted in the left figure, we confirm that the error bound is rather sharp, almost indistinguishable from the true error even if we use a constant $\alpha = 1$ for the lower bound in (4.12) and (4.14).

Figure 5.12 reports the reduced basis error bound during the hierarchical construction process for both the weighted scheme and non weighted scheme. Large oscillation of the worst error bound evaluated at the nodes corresponding to the current active index can be observed for both cases. Both of them decrease to the prescribed tolerance $\epsilon_t = 10^{-11}$ but with different number of bases. In fact, the probability density ρ in (4.14) becomes very small when the node is far away from the center, thus gives rise to very small weighted a posteriori error bound and early stop of the algorithm with less bases. Moreover, this test also demonstrates that the total number of reduced bases is much smaller

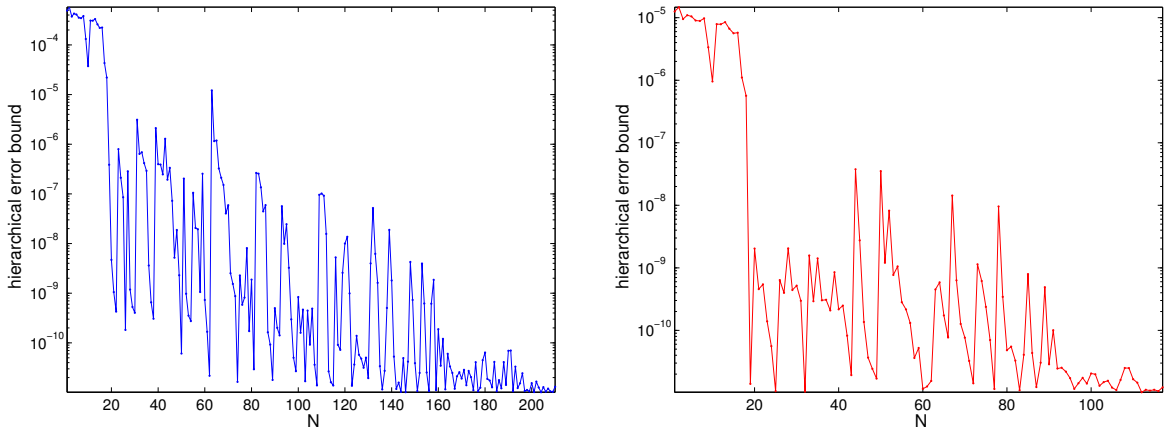

Figure 5.12: Left: the a posteriori error bound (4.12); right: the weighted a posteriori error bound (4.14) during the hierarchical construction of the generalized sparse grid by Algorithm 2.

than the total number of constructed nodes, thus efficiently alleviate the entire computational cost.

In the second test, we consider a high-dimensional heat diffusion problem with 100 thermal blocks. The conductivity coefficient is

$$a(x,y) = \sum_{k=1}^{K} \chi_{D_k}(x) 10^{c_k(y_k - 0.5)} \tag{5.7}$$

where $y_k \in [0,1], 1 \le k \le K$ with $K = 100$, are independent and uniformly distributed random variables; $c_k, 1 \le k \le K$, are taken similarly to the second test of section 5.4 in separating the dimensions into two scales: we randomly select $2\sqrt{K}$ dimensions and set $c_k = 4y_k^0$ in these dimensions and $c_k = 10^{-4} \times 4y_k^0$ in the other dimensions, being $y_k^0 \in [0,1], 1 \le k \le K$, samples drawn from uniform distributed random variable. We set the error tolerance for the reduced basis space construction as $\epsilon_t = 10^{-8}$ in the greedy Algorithm 3 and the maximum number of nodes as $M = 10^n, 0 \le n \le 5$ for the hierarchical construction of the generalized sparse grid in Algorithm 2, which result in 161 bases in the reduced basis space. On the right, the integration error computed with different number of nodes are shown, which decays with a rate larger than 1, demonstrating that the dimension-adaptive hierarchical approximation converges much faster than Monte Carlo method for this high-dimensional uncertainty quantification problem.

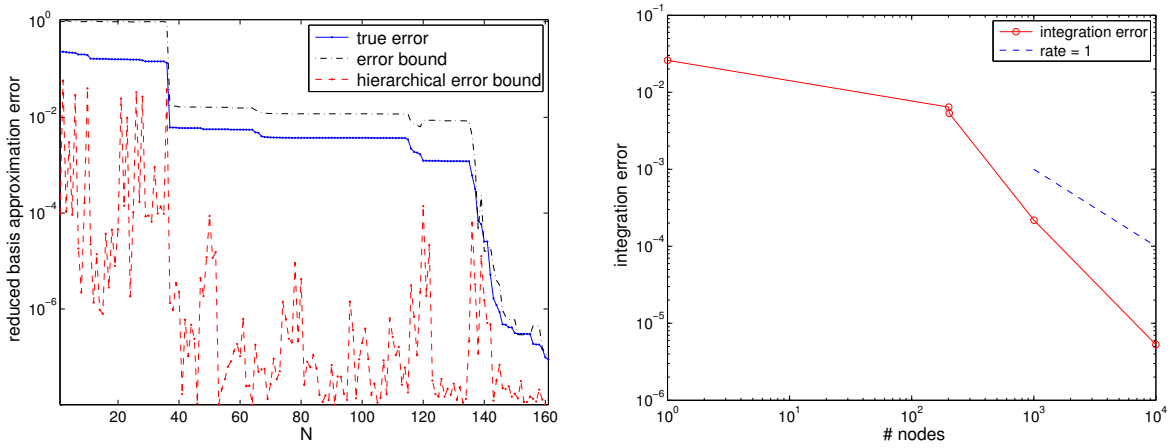

Figure 5.13: Left: true error and error bound of reduced basis approximation; right: integration error.

Figure 5.13 displays both the reduced basis approximation error and the integration error. On the left, the true error and the error bound (in maximum norm) evaluated at 100 randomly selected samples at different number of reduced bases confirm the effectivity of the a posteriori error bound. The a posteriori error bounds at the selected reduced basis samples, most of which are chosen at the beginning of the hierarchical construction process, decrease in an oscillating way to the error tolerance and remain smaller than the maximum error bounds at the 100 samples. Figure 5.14 depicts the effective dimensions and varied importance of different dimensions indicated by the prescribed parameters (on the left) and the level of the generalized sparse grid in different dimensions (on the right). From this figure, we can observe that all the dimensions in the effective scale represented by the characteristic function $\chi_d(k), 1 \leq k \leq K$, (on the left) are correctly identified with the grid level i_k equal or larger than 4 (on the right), and the dimensions in the ineffective scale are approximated mostly by the grid level $i_k = 1 + 1$. Moreover, the varied importance of different dimensions in each scale is also successfully identified as shown in Figure 5.14, where a larger value of y_k^0 leads to a relatively deeper grid level.

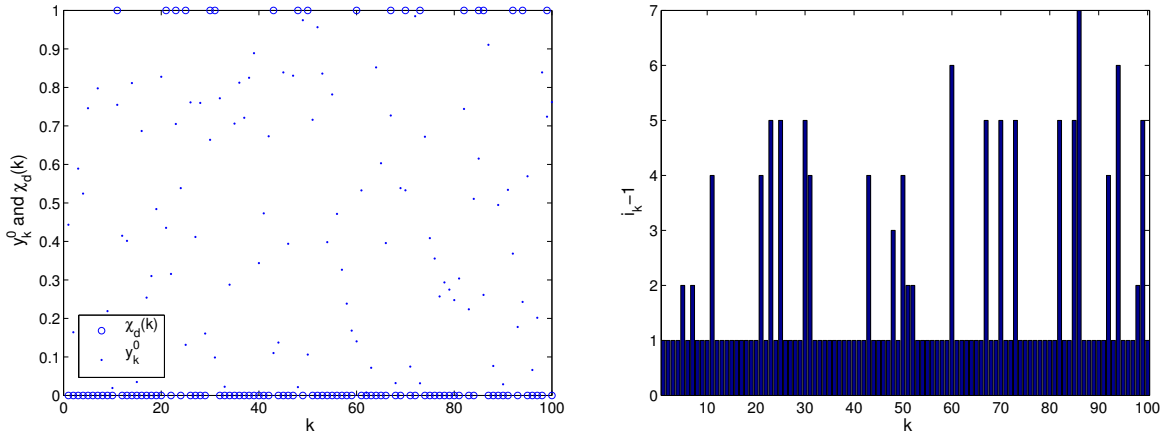

Figure 5.14: Left: true error and error bound of reduced basis approximation; right: integration error.

5.6 Groundwater flow through porous medium

This example is devoted to the study of groundwater flow through porous medium described by Darcy's law: find the pressure field $p \in D \times \Gamma$ such that the following equations hold

$$\begin{cases} -\nabla(a\nabla p) &= 0 \quad \text{in } D, \\ p &= 1 \quad \text{on } \partial D_4, \\ p &= 0 \quad \text{on } \partial D_2, \\ a\nabla p \cdot \mathbf{n} &= 0 \quad \text{on } \partial D_1 \cup \partial D_3, \end{cases} \tag{5.8}$$

where the physical domain is the two dimensional square $D = (0,1)^2$, as shown in Figure 5.15, with left and right boundaries ($\partial D_2 \cup \partial D_4$) prescribed of Dirichlet boundary conditions, and the upper and lower boundaries ($\partial D_1 \cup \partial D_3$) homogeneous Neuman boundary conditions. The permeability of the porous medium is given by the random field (with $x = (x_1, x_2)$)

$$a(x,y) = \mathbb{E}[a] + \left(\frac{\sqrt{\pi}L}{2}\right)^{1/2} y_1 + \sum_{k=1}^{K} \sqrt{\lambda_k}\left(\sin(k\pi x_1)y_{2k} + \cos(k\pi x_1)y_{2k+1}\right), \tag{5.9}$$

which is a truncated Karhunen-Loève expansion of a Gauss covariance kernel $\exp(-(x_1 - x_1')^2/L^2)$ with correlation length L [47]. The eigenvalues $\lambda_k, 1 \leq k \leq K$, of this kernel decay exponentially as

$$\lambda_k = \sqrt{\pi} L \exp\left(-\frac{(k\pi L)^2}{4}\right), \tag{5.10}$$

and the random variables $y_k, 1 \leq k \leq 2K+1$, are assumed to be independent and obey uniform distribution taking values in $[-\sqrt{3}, \sqrt{3}]$ in order to guarantee that a is positive. The quantity of interest is

$$s(y) := L(p; y) = \int_{D_d} a(x, y) \partial_{x_1} p(x, y) dx, \tag{5.11}$$

where the disk region D_d has center $(0.75, 0.5)$ and radius 0.2, see Figure 5.15. This quantity is not compliant with the right hand side of equation $(5.8)_1$. Therefore, we adopt the primal-dual approach introduced in section 4.5. We first write the weak formulation of the Darcy equation (5.8) as: find $p \in H^1(D)$ such that

$$A(p, q; y) = 0 \quad \forall q \in H^1_{dir}(D), \tag{5.12}$$

where $H^1_{dir}(D) := \{q \in H^1(D) : q = 0 \text{ on } \partial D_2 \cup \partial D_4\}$ and the bilinear form A is given by

$$A = \sum_{k=0}^{2K+1} A_k(p, q) y_k, \tag{5.13}$$

being A_k defined corresponding to the terms in the expansion of the permeability coefficient a in (5.9) and $y_0 = 1$ for notational convenience. The dual problem associated with the primal problem (5.12) for the quantity of interest s is formulated as: find $\varphi \in H^1_{dir}$ such that

$$A(q, \varphi; y) = -L(q; y) \quad \forall q \in H^1_{dir}(D). \tag{5.14}$$

We construct reduced basis space $X^{pr}_{N_{pr}}$ with N_{pr} bases and $X^{du}_{N_{du}}$ with N_{du} bases to approximate the primal and dual weak problems (5.12) and (5.14) and define the residual of each problem as

$$R^{pr}(q; y) = -A(p_{N_{pr}}, q; y) \text{ and } R^{du}(q; y) = -L(q; y) - A(q, \varphi_{N_{du}}; y), \tag{5.15}$$

where $p_{N_{pr}}$ and $\varphi_{N_{du}}$ are the reduced basis approximations of the primal and dual solutions, respectively. We apply piecewise finite element basis to approximate these solutions in the physical space and denote the approximation space as $X \subset H^1_{dir}(D)$ and its dual as X'. After solving the primal and dual reduced basis problems, we can approximate the quantity of interest s defined in (5.11) by

$$s_N(y) = L(p_{N_{pr}}(y); y) - R^{pr}(\varphi_{N_{du}}(y); y), \tag{5.16}$$

whose error can be bounded as (see details in [61])

$$|s(y) - s_N(y)| \leq \triangle^s_N(y) := \frac{||R^{pr}(\cdot; y)||_{X'} ||R^{du}(\cdot; y)||_{X'}}{\alpha(y)}. \tag{5.17}$$

For the approximation in physical space, we use piecewise linear finite element basis on a regular triangular mesh with 17361 vertices, leading to a relatively large-scale algebraic system. We run Algorithm 2 for the dimension-adaptive hierarchical construction of the generalized space grid with integration error indicator (3.23) at a series of maximum number of nodes $10^n, 0 \leq n \leq 5$. The error tolerance for the reduced basis construction for approximating the non-compliant quantity of interest s is set as $\epsilon_t = 10^{-8}$. The a posteriori error bound (5.17) can be efficiently evaluated by an offline-online decomposition procedure for both the primal and dual problems with error tolerance $\epsilon_t = 10^{-4}$ for both problems. We set the correlation length $L = 1/16$ in (5.10) and K as $8, 16, 32, 64$, which lead to $17, 33, 65, 129$ dimensions taking $59\%, 89\%, 99\%$ and 100% percent of the total randomness measured by the L^∞-norm of the coefficient in (5.9). A set of typical solutions of the primal problem (5.12) and

dual problem (5.14) at a randomly selected sample are depicted in Figure 5.15 (middle and right), where the dual solution, with evident bigger values near the disk D_d plays the role to correct the reduced basis approximation of the quantity of interest s_N by formula (5.16).

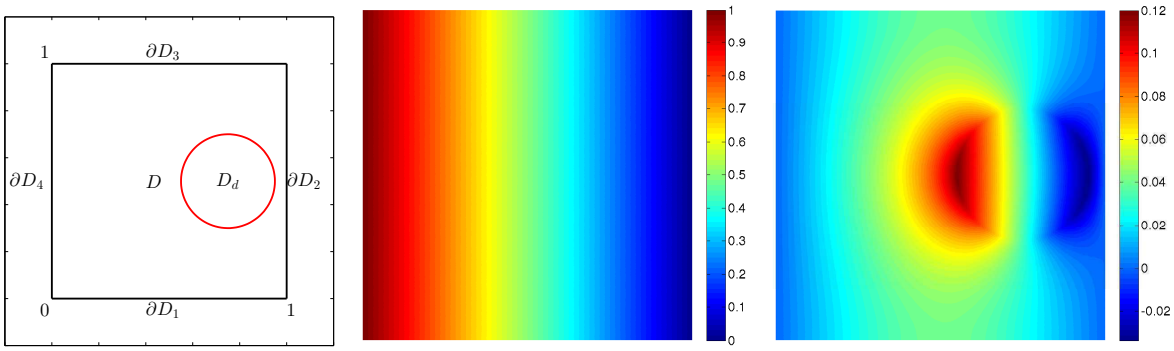

Figure 5.15: Left: physical domain and boundaries; middle and right: primal and dual solutions.

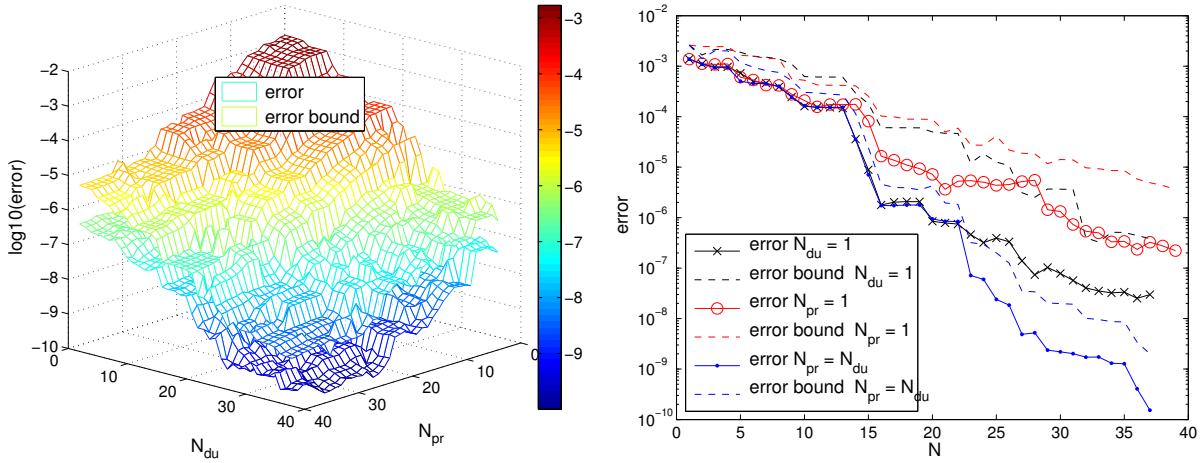

Figure 5.16: Left: reduced basis approximation error and error bound w.r.t. the number of primal bases N_{pr} and the number of dual bases N_{du}; right: three different settings of N_{pr} and N_{du}.$K = 8$.

In the 17 dimensional case ($K = 8$), 37 primal bases and 39 dual bases are constructed. We test the convergence of the worst reduced basis approximation error with respect to the number of primal bases and dual bases computed with 100 randomly selected samples, which is displayed in Figure 5.16. From the left figure, we can observe that both the approximation error and the error bound decrease with growing number of primal and dual bases, leading to quadratically fast decrease with N_{pr} and N_{du} increasing simultaneously as shown in the right figure on the path $N_{pr} = N_{du}$. Moreover, the error bound shown in this figure is rather sharp (close to the real approximation error), demonstrating the efficiency of the primal-dual approach using the a posteriori error bound (5.17). The interpolation errors by the hierarchical interpolation formula (3.20) with 10^5 interpolation nodes are evaluated at the same test samples, where the worst approximation error is 4.0603×10^{-5}, much larger than that of the reduced basis approximation error 1.3333×10^{-10}. This large difference is due to fact that the interpolation approach adopts Lagrange basis to approximate the pointwise quantities blind to the underlying PDE model, while the reduced basis approach performs the pointwise evaluation by solving the underlying PDE model with cheap cost in the reduced framework. Therefore, we always use the reduced basis approximation to evaluate pointwise value of quantity of interest s.

The worst approximation error and integration error for the cases $K = 8, 16, 32, 64$, corresponding

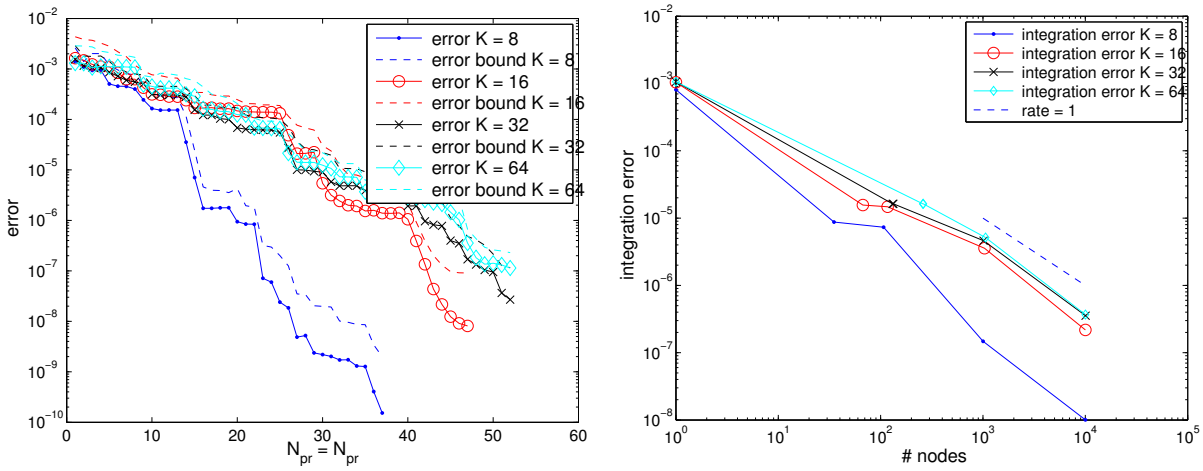

Figure 5.17: Left: reduced basis approximation error and error bound with $N_{pr} = N_{du}$; right: integration error. $K = 8, 16, 32, 64$, corresponding to 17, 33, 65, 129 dimensional problems.

to $17, 33, 65, 129$ dimensional problems, are reported in Figure 5.17. The number of primal and dual bases increases with the dimension in order to achieve the same accuracy of approximation for pointwise evaluation and integration. However, the increase is rather small when the dimension becomes high because the important dimensions have been captured by the reduced basis approximation and dimension-adaptive hierarchical integration in the first few dimensions, and all the other dimensions play negligible role in contributing to the approximation error. It is worth to point out the remarkable fact that only a few tens (about 50) of reduced bases have been constructed to approximate the high dimension uncertainty quantification problems as shown in this example, thus requiring only a few tens of full solves of the underlying PDE model compared to a really large number (10^5 in this example) of full solves that would be needed without using the reduced basis method. Furthermore, as shown in the right of Figure 5.17 that the integration error converges with rate larger than 1, which demonstrates that the adaptive and reduced computational strategy for integration in high dimensions is very promising.

6 Concluding remarks

In this paper we have developed an adaptive reduced computational framework for solving high-dimensional uncertainty quantification problems. Two critical computational challenges were identified and illustrated for various uncertainty quantification problems: curse-of-dimensionality and heavy computational burden. In order to tackle the first challenge, we adopted the approach for dimension-adaptive tensor product integration and developed a verified algorithm based on generalized sparse grid construction to deal with one drawback of this approach - the stagnation phenomenon, and designed different error indicators suitable for integration and interpolation problems based on the hierarchical surpluses. To overcome the second challenge, we developed an adaptive and weighted reduced basis method, using an adaptive greedy algorithm in combination with the dimension-adaptive hierarchical grid construction and a weighted a posteriori error bound to alleviate the computational cost in building the reduced basis space. Different techniques for extensions to more general PDE models are summarized for application of the reduced basis method in more general UQ problems. The numerical experiments demonstrated that Algorithm 2 worked effectively in getting rid of the stagnation phenomenon and in automatically detecting the importance and interaction of different dimensions, which converged faster than the Monte Carlo and quasi Monte Carlo methods for high-dimensional integration problems. Moreover, the integration error indicator incorporating hierarchical surpluses, work contributions as well as quadrature weights was proved to be very efficient for UQ problems with arbitrary probability measures. As for pointwise evaluation of output of interest de-

pending on PDE solution, the reduced basis approximation certified by the a posteriori error bound was demonstrated to be more accurate than the interpolation scheme based on Lagrange polynomials, one kind of dictionary bases without taking into account the underlying PDE models. Furthermore, only a few bases, a few hundreds (about 100 - 200) for heat diffusion in thermal blocks and a few tens (about 40 - 50) for groundwater flow through porous medium compared to 10^5 full solves, were constructed by the reduced basis method in order to achieve great accuracy for the high-dimensional approximation problems. This reduction will dramatically alleviate the prohibitive computational effort to the affordable level in solving large-scale PDE models (with large degrees of freedom) that consume considerable computational power.

Several further topics are worth to be investigated in applying the adaptive and reduced computational framework to solve high-dimensional uncertainty quantification problems. The first is that low regularity points may exist in the high-dimensional space, for instance, the points featuring discontinuity or singularity. Therefore, efficient low regularity detection algorithms need to be incorporated in this framework, e.g. by checking the pointwise hierarchical surpluses instead of an averaged or maximum value at one index [42]. In addition to the detection algorithm, we remark that the reduced basis approximation may essentially get rid of the low regularity problems since it does not apply any family of dictionary bases but project the new solution into the reduced basis space spanned by solutions at some selected samples [12]. Another research topic is to develop more specific and goal-oriented model order reduction techniques in order to circumvent the "irreducible" PDE models, such as locally supported traveling waves, compressible flows that feature shocks, and so on. Last but not least, when the effective dimensions become so high that the dimension-adaptive quadrature rule converges too slow, we have to turn to other approaches, such as Monte Carlo method. Since the reduced basis method is still applicable for Monte Carlo method, the adaptation may be carried out for sampling set with successive enrichment of new samples and elimination of well approximated samples, as already done in [12] for risk analysis.

Acknowledgement: We acknowledge the use of the Matlab packages *rbMIT* developed by the group of Prof. Anthony Patera at MIT for reduced basis method, *MLife* previously developed by Prof. Fausto Saleri from MOX, Politecnico di Milano for finite element solver and *spinterp* by Dr. Andreas Klimke from Universität Stuttgart for sparse grid interpolation.

References

[1] I. Babuška, F. Nobile, and R. Tempone. A stochastic collocation method for elliptic partial differential equations with random input data. *SIAM Journal on Numerical Analysis*, 45(3):1005–1034, 2007.

[2] I. Babuška, F. Nobile, and R. Tempone. A stochastic collocation method for elliptic partial differential equations with random input data. *SIAM Review*, 52(3):317, 2010.

[3] I. Babuška, R. Tempone, and G.E. Zouraris. Galerkin finite element approximations of stochastic elliptic partial differential equations. *SIAM Journal on Numerical Analysis*, 42(2):800–825, 2005.

[4] M. Barrault, Y. Maday, N.C. Nguyen, and A.T. Patera. An empirical interpolation method: application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus Mathematique, Analyse Numérique*, 339(9):667–672, 2004.

[5] L. Biegler, G. Biros, and O. Ghattas. *Large-scale inverse problems and quantification of uncertainty*. Wiley, 2011.

[6] M. Bieri and C. Schwab. Sparse high order FEM for elliptic sPDEs. *Computer Methods in Applied Mechanics and Engineering*, 198(13):1149–1170, 2009.

[7] G. Blatman and B. Sudret. Sparse polynomial chaos expansions and adaptive stochastic finite elements using a regression approach. *Comptes Rendus Mécanique*, 336(6):518–523, 2008.

[8] S. Boyaval, C. Le Bris, T. Lelièvre, Y. Maday, N.C. Nguyen, and A.T. Patera. Reduced basis techniques for stochastic problems. *Archives of Computational Methods in Engineering*, 17:435–454, 2010.

[9] H.J. Bungartz and M. Griebel. Sparse grids. *Acta Numerica*, 13(1):147–269, 2004.

[10] G.T. Buzzard and D. Xiu. Variance-based global sensitivity analysis via sparse-grid interpolation and cubature. *Communications in Computational Physics*, 9:542–67, 2011.

[11] C. Canuto, M.Y. Hussaini, A. Quarteroni, and T.A. Zang. *Spectral Methods: Fundamentals in Single Domains*. Springer, 2006.

[12] P. Chen and A. Quarteroni. Accurate and efficient evaluation of failure probability for partial differential equations with random input data. *Computer Methods in Applied Mechanics and Engineering*, 267(0):233–260, 2013.

[13] P. Chen and A. Quarteroni. Weighted reduced basis method for stochastic optimal control problems with elliptic PDE constraints. *submitted*, 2013.

[14] P. Chen, A. Quarteroni, and G. Rozza. Multilevel and weighted reduced basis method for stochastic optimal control problems constrained by stokes equations. *submitted*, 2013.

[15] P. Chen, A. Quarteroni, and G. Rozza. Simulation-based uncertainty quantification of human arterial network hemodynamics. *International Journal for Numerical Methods in Biomedical Engineering*, 29(6):698–721, 2013.

[16] P. Chen, A. Quarteroni, and G. Rozza. Stochastic optimal Robin boundary control problems of advection-dominated elliptic equations. *SIAM Journal on Numerical Analysis*, 51(5):2700 – 2722, 2013.

[17] P. Chen, A. Quarteroni, and G. Rozza. A weighted empirical interpolation method: A priori convergence analysis and applications. *accepted in ESAIM: Mathematical Modelling and Numerical Analysis, EPFL, MATHICSE Report 05,*, 2013.

[18] P. Chen, A. Quarteroni, and G. Rozza. A weighted reduced basis method for elliptic partial differential equations with random input data. *SIAM Journal on Numerical Analysis*, 51(6):3163 – 3185, 2013.

[19] Y. Chen, J.S. Hesthaven, Y. Maday, and J. Rodríguez. Certified reduced basis methods and output bounds for the harmonic Maxwell's equations. *SIAM Journal on Scientific Computing*, 32(2):970–996, 2010.

[20] J. Dick, F.Y. Kuo, and I.H. Sloan. High-dimensional integration–the Quasi-Monte Carlo way. *Acta Numerica*, 22:133–288, 2013.

[21] M. Drohmann, B. Haasdonk, and M. Ohlberger. Reduced basis approximation for nonlinear parametrized evolution equations based on empirical operator interpolation. *SIAM Journal on Scientific Computing*, 34(2):A937–A969, 2012.

[22] R. Durrett. *Probability: theory and examples*. Cambridge University Press, 2010.

[23] L.C. Evans. *Partial Differential Equations, Graduate Studies in Mathematics, Vol. 19, American Mathematical Society*. 2009.

[24] C. Feuersänger. *Sparse Grid Methods for Higher Dimensional Approximation*. Dissertation, Institut für Numerische Simulation, Universität Bonn, September 2010.

[25] G.S. Fishman. *Monte Carlo: Concepts, Algorithms, and Applications*. Springer, 1996.

[26] J. Foo and G.E. Karniadakis. Multi-element probabilistic collocation method in high dimensions. *Journal of Computational Physics*, 229(5):1536–1557, 2010.

[27] P. Frauenfelder, C. Schwab, and R.A. Todor. Finite elements for elliptic problems with stochastic coefficients. *Computer methods in applied mechanics and engineering*, 194(2-5):205–228, 2005.

[28] Z. Gao and J.S. Hesthaven. On ANOVA expansions and strategies for choosing the anchor point. *Applied Mathematics and Computation*, 217(7):3274–3285, 2010.

[29] T. Gerstner and M. Griebel. Dimension–adaptive tensor–product quadrature. *Computing*, 71(1):65–87, 2003.

[30] R.G. Ghanem and P.D. Spanos. *Stochastic Finite Elements: a Spectral Approach*. Dover Civil and Mechanical Engineering, Courier Dover Publications, 2003.

[31] M.A. Grepl, Y. Maday, N.C. Nguyen, and A.T. Patera. Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 41(03):575–605, 2007.

[32] M.A. Grepl and A.T. Patera. A posteriori error bounds for reduced-basis approximations of parametrized parabolic partial differential equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 39(01):157–181, 2005.

[33] M. Griebel. Sparse grids and related approximation schemes for higher dimensional problems. In *Proceedings of the conference on Foundations of Computational Mathematics*, Santander,Spain, 2005.

[34] M. Gunzburger and A. Labosvky. An efficient and accurate numerical method for high-dimensional stochastic partial differential equations. *submitted*.

[35] B. Haasdonk and M. Ohlberger. Reduced basis method for finite volume approximations of parametrized linear evolution equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 42(02):277–302, 2008.

[36] J.S. Hesthaven and S. Zhang. On the use of ANOVA expansions in reduced basis methods for high-dimensional parametric partial differential equations. *Brown Division of Applied Math Scientific Computing Tech Report*, 2011.

[37] M. Holtz. *Sparse Grid Quadrature in High Dimensions with Applications in Finance and Insurance*. Dissertation, Institut für Numerische Simulation, Universität Bonn, 2008.

[38] D.B.P Huynh, G. Rozza, S. Sen, and A.T. Patera. A successive constraint linear optimization method for lower bounds of parametric coercivity and inf-sup stability constants. *Comptes Rendus Mathematique, Analyse Numérique*, 345(8):473–478, 2007.

[39] A. Klimke. *Uncertainty modeling using fuzzy arithmetic and sparse grids. Universität Stuttgart*. PhD thesis, Universität Stuttgart, 2006.

[40] T. Lassila, A. Quarteroni, and G. Rozza. A reduced basis model with parametric coupling for fluid-structure interaction problems. *SIAM Journal on Scientific Computing*, 34(2):1187–1213, 2012.

[41] J. Li and D. Xiu. Evaluation of failure probability via surrogate models. *Journal of Computational Physics*, 229(23):8966–8980, 2010.

[42] X. Ma and N. Zabaras. An adaptive high-dimensional stochastic model representation technique for the solution of stochastic partial differential equations. *Journal of Computational Physics*, 229(10):3884–3915, 2010.

[43] Y. Maday, A.T. Patera, and G. Turinici. A priori convergence theory for reduced-basis approximations of single-parameter elliptic partial differential equations. *Journal of Scientific Computing*, 17(1):437–446, 2002.

[44] G. Migliorati, F. Nobile, E. Von Schwerin, and R. Tempone. Approximation of quantities of interest in stochastic PDEs by the random discrete L^2 projection on polynomial spaces. *SIAM Journal on Scientific Computing*, 35(3):A1440–A1460, 2013.

[45] P.B. Nair and A.J. Keane. Stochastic reduced basis methods. *AIAA journal*, 40(8):1653–1664, 2002.

[46] F. Nobile, R. Tempone, and C.G. Webster. An anisotropic sparse grid stochastic collocation method for partial differential equations with random input data. *SIAM Journal on Numerical Analysis*, 46(5):2411–2442, 2008.

[47] F. Nobile, R. Tempone, and C.G. Webster. A sparse grid stochastic collocation method for partial differential equations with random input data. *SIAM Journal on Numerical Analysis*, 46(5):2309–2345, 2008.

[48] A.K. Noor and J.M. Peters. Reduced basis technique for nonlinear analysis of structures. *AIAA Journal*, 18(4):455–462, 1980.

[49] A. Nouy. A generalized spectral decomposition technique to solve a class of linear stochastic partial differential equations. *Computer Methods in Applied Mechanics and Engineering*, 196(45-48):4521–4537, 2007.

[50] A. Nouy. Recent developments in spectral stochastic methods for the numerical solution of stochastic partial differential equations. *Archives of Computational Methods in Engineering*, 16(3):251–285, 2009.

[51] A. Nouy. Proper generalized decompositions and separated representations for the numerical solution of high dimensional stochastic problems. *Archives of Computational Methods in Engineering*, 17(4):403–434, 2010.

[52] A.T. Patera and G. Rozza. Reduced basis approximation and a posteriori error estimation for parametrized partial differential equations Version 1.0. *Copyright MIT, http://augustine.mit.edu*, 2007.

[53] T.N.L. Patterson. The optimum addition of points to quadrature formulae. *Mathematics of Computation*, 22(104):847–856, 1968.

[54] A. Quarteroni. *Numerical Models for Differential Problems*. Springer, 2nd ed., 2013.

[55] A. Quarteroni and G. Rozza. Numerical solution of parametrized Navier–Stokes equations by reduced basis methods. *Numerical Methods for Partial Differential Equations*, 23(4):923–948, 2007.

[56] A. Quarteroni, G. Rozza, and A. Manzoni. Certified reduced basis approximation for parametrized partial differential equations and applications. *Journal of Mathematics in Industry*, 1(1):1–49, 2011.

[57] A. Quarteroni, R. Sacco, and F. Saleri. *Numerical Mathematics*. Springer, 2007.

[58] A. Quarteroni and A. Valli. *Numerical Approximation of Partial Differential Equations*. Springer, 1994.

[59] G. Rozza. *Shape design by optimal flow control and reduced basis techniques: Applications to bypass configurations in haemodynamics*. PhD thesis, EPFL, 2005.

[60] G. Rozza, D.B.P. Huynh, and A. Manzoni. Reduced basis approximation and a posteriori error estimation for Stokes flows in parametrized geometries: roles of the inf-sup stability constants. *Numerische Mathematik*, pages 1–38, 2013.

[61] G. Rozza, D.B.P. Huynh, and A.T. Patera. Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations. *Archives of Computational Methods in Engineering*, 15(3):229–275, 2008.

[62] G. Rozza and K. Veroy. On the stability of the reduced basis method for stokes equations in parametrized domains. *Computer methods in applied mechanics and engineering*, 196(7):1244–1260, 2007.

[63] A. Saltelli, K. Chan, and E.M. Scott. *Sensitivity analysis*, volume 134. Wiley New York, 2000.

[64] C. Schwab and A.M. Stuart. Sparse deterministic approximation of bayesian inverse problems. *Inverse Problems*, 28(4):045003, 2012.

[65] C. Schwab and R. A. Todor. Karhunen-Loève approximation of random fields by generalized fast multipole methods. *Journal of Computational Physics*, 217(1):100–122, 2006.

[66] C. Schwab and R.A. Todor. Sparse finite elements for elliptic problems with stochastic loading. *Numerische Mathematik*, 95(4):707–734, 2003.

[67] S.A. Smolyak. Quadrature and interpolation formulas for tensor products of certain classes of functions. In *Doklady Akademii Nauk SSSR*, volume 4, pages 240–243, 1963.

[68] I.M. Sobol. Theorems and examples on high dimensional model representation. *Reliability Engineering & System Safety*, 79(2):187–193, 2003.

[69] H. Tiesler, R.M. Kirby, D. Xiu, and T. Preusser. Stochastic collocation for optimal control problems with stochastic PDE constraints. *SIAM Journal on Control and Optimization*, 50(5):2659–2682, 2012.

[70] L.N. Trefethen. Is Gauss quadrature better than Clenshaw-Curtis? *SIAM Review*, 50(1):67–87, 2008.

[71] J. Walsh. An introduction to stochastic partial differential equations. *École d'Été de Probabilités de Saint Flour XIV-1984*, pages 265–439, 1986.

[72] D. Xiu. Fast numerical methods for stochastic computations: a review. *Communications in Computational Physics*, 5(2-4):242–272, 2009.

[73] D. Xiu and J.S. Hesthaven. High-order collocation methods for differential equations with random inputs. *SIAM Journal on Scientific Computing*, 27(3):1118–1139, 2005.

[74] D. Xiu and G.E. Karniadakis. The Wiener-Askey polynomial chaos for stochastic differential equations. *SIAM Journal on Scientific Computing*, 24(2):619–644, 2003.

[75] N. Zabaras and B. Ganapathysubramanian. A scalable framework for the solution of stochastic inverse problems using a sparse grid collocation approach. *Journal of Computational Physics*, 227(9):4697–4735, 2008.

MOX Technical Reports, last issues

Dipartimento di Matematica "F. Brioschi",
Politecnico di Milano, Via Bonardi 9 - 20133 Milano (Italy)

09/2014 Chen, P.; Quarteroni, A.
A new algorithm for high-dimensional uncertainty quantification problems based on dimension-adaptive and reduced basis methods

08/2014 Cattaneo, L; Zunino, P.
A computational model of drug delivery through microcirculation to compare different tumor treatments

07/2014 Agasisti, T.; Ieva, F.; Paganoni, A.M.
Heterogeneity, school-effects and achievement gaps across Italian regions: further evidence from statistical modeling

06/2014 Benzi, M.; Deparis, S.; Grandperrin, G.; Quarteroni, A.
Parameter estimates for the relaxed dimensional factorization preconditioner and application to hemodynamics

05/2014 Rozza, G.; Koshakji, A.; Quarteroni, A.
Free Form Deformation Techniques Applied to 3D Shape Optimization Problems

04/2014 Palamara, S.; Vergara, C.; Catanzariti, D.; Faggiano, E.; Centonze, M.; Pangrazzi, C.; Maines, M.; Quarteroni, A.
Patient-specific generation of the Purkinje network driven by clinical measurements: The case of pathological propagations

03/2014 Kashiwabara, T.; Colciago, C.M.; Dede, L.; Quarteroni, A.
Well-posedness, regulariy, and convergence analysis of the Finite Element approximation of a Generalized Robin boundary value problem

02/2014 Antonietti, P.F.; Sarti, M.; Verani, M.
Multigrid algorithms for high order discontinuous Galerkin methods

01/2014 Secchi, P.; Vantini, S.; Zanini, P.
Hierarchical Independent Component Analysis: a multi-resolution non-orthogonal data-driven basis

67/2013 Canuto, C.; Simoncini, V.; Verani, M.
On the decay of the inverse of matrices that are sum of Kronecker products