Journal of Computational Physics 298 (2015) 176–193

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

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

Peng Chen [a,*], Alfio Quarteroni [b]

[a] Seminar for Applied Mathematics, ETH Zürich, Rämistrasse 101, CH-8092 Zürich, Switzerland
[b] 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

ARTICLE INFO

Article history:
Received 11 February 2014
Received in revised form 10 April 2015
Accepted 2 June 2015
Available online 16 June 2015

Keywords:
Uncertainty quantification
Curse of dimensionality
Generalized sparse grid
Hierarchical surpluses
Reduced basis method
Adaptive greedy algorithm
Weighted a posteriori error bound

ABSTRACT

In this work we develop an adaptive and reduced computational algorithm based on dimension-adaptive sparse grid approximation and reduced basis methods 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 [16] 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 (PDE), we adopt a weighted reduced basis method [7] and develop an adaptive greedy algorithm in combination with the previous verified algorithm for efficient construction of an accurate reduced basis approximation. The efficiency and accuracy of the proposed algorithm are demonstrated by several numerical experiments.

1. Introduction

Many different computational methods have been proposed and developed during the last few decades to solve uncertainty quantification problems. Among the most widely used is the sampling based Monte Carlo method and its various accelerated versions [13,11], which are straightforward for implementation although they are 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 [17,35,2,32]. Another efficient sampling based method, the stochastic collocation method [34,1], 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 [34,24] are applied to reduce the total number of collocation samples. More recently, model order reduction techniques, including the reduced basis method [4,12,18,8] and the proper generalized decomposition method [25], have been developed. In both cases, the stochastic solution is expanded with respect to a few basis functions that are constructed offline and represent high-fidelity solutions of the underlying PDE. When the dimension of the uncertainties becomes high, the number of isotropic projection basis functions or collocation nodes may grow exponentially fast, one computational challenge known as “curse of

[*] Corresponding author.
E-mail addresses: peng.chen@sam.math.ethz.ch, cpempire@gmail.com (P. Chen), alfio.quarteroni@epfl.ch (A. Quarteroni).

http://dx.doi.org/10.1016/j.jcp.2015.06.006
0021-9991/© 2015 Elsevier Inc. All rights reserved.

dimensionality". 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) involved 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.

A strategy to tackle the curse of dimensionality is to take advantage of the sparsity – the importance of different dimensions and their interaction/combination is very different, so that only a limited number of dimensions play an effective role [9]. This idea has led to the development of the weighted function space based quasi Monte Carlo method [11], a priori and a posteriori analysis based anisotropic sparse grid construction [23], (Sobol) decomposition of function based techniques such as ANOVA (analysis of variance) [15], HDMR (high-dimensional model representation) [22]. Another recently developed method under the name of dimension-adaptive tensor-product integration [16] 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 choices 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 an unnecessarily 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. Moreover, 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 [16] and [21], 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 through the whole grid construction. Rather, 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. 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 time.

The verification remedy has not yet been studied in the literature, neither 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 (see e.g. [30,27]), which has been used in combination with ANOVA in [19], and develop an adaptive and weighted algorithm in combination with the verified hierarchical approximation. Based on this idea and using the reduced basis method, 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 basis functions [7]. This proves to be more efficient with much less basis functions in achieving the same approximation accuracy as the a posteriori error bound without incorporating the weight. In the end, an adaptive and reduced computational algorithm is developed in efficiently and accurately solving high-dimensional UQ problems that feature sparsity and reducibility. A series of numerical experiments featuring various properties for both functions and the underlying PDE are carried out to demonstrate the efficiency and accuracy of our method and compare its computational performance to several other techniques.

The paper is organized as follows. We briefly introduce uncertainty quantification problems in Section 2. Section 3 is devoted to the development of the verified dimension-adaptive hierarchical approximation based on generalized sparse grid construction. In Section 4, the adaptive and weighted reduced basis method is presented, 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. Numerical experiments are presented in Section 5 to demonstrate the accuracy and efficiency of the proposed computational algorithm. We close the paper by drawing some conclusions and perspectives in the last section.

2. Problem setting

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 probability density function of a random variable $Y : \Omega \to \Gamma \subset \mathbb{R}$, being Γ the image of Y, is denoted as ρ. For any positive integer $k \in \mathbb{N}_+$, the k-th moment of Y is defined as

$$\mathbb{E}\left[Y^k\right] = \int_\Gamma y^k \rho(y) dy. \tag{1}$$

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

$$\begin{cases} \mathcal{L}(u; x, \omega) = f(x, \omega) & x \in D, \\ \mathcal{B}(u; x, \omega) = g(x, \omega) & x \in \partial D. \end{cases} \tag{2}$$

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 the external loading and the boundary condition, respectively. The uncertainties, so-called random inputs, may arise from material coefficients, the computational geometry, the external loading and/or the boundary condition. We assume that the random inputs in $\mathcal{L}$, $\mathcal{B}$, f and/or g satisfy certain regularity conditions in both the physical space and the probability space [2], so that the problem (2) is well posed. Another important assumption, which enables the application of the stochastic computational methods introduced in this work, is the so called "finite dimensional noise assumption" [1]. In short, the random inputs live in a finite dimensional probability space in the sense that there exist $K \in \mathbb{N}_+$ random variables $Y_k : \Omega \to \mathbb{R}$, $k = 1, \dots, K$, such that the uncertainties in (2) depend on ω only through $Y_1(\omega), \dots, Y_K(\omega)$. Therefore, $u(x, \omega) = u(x, Y_1(\omega), \dots, Y_K(\omega))$. 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 $k = 1, \dots, K$. 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 problem (2) can be recast as a parametric PDE: find $u : D \times \Gamma \to \mathbb{R}$ such that for ρ-a.e. $y \in \Gamma$

$$\begin{cases} \mathcal{L}(u; x, y) = f(x, y) & x \in D, \\ \mathcal{B}(u; x, y) = g(x, y) & x \in \partial D. \end{cases} \tag{3}$$

A simple example is the linear elliptic problem: find $u : D \times \Gamma \to \mathbb{R}$ such that for ρ-a.e. $y \in \Gamma$

$$\begin{cases} -\nabla(a(x, y)\nabla u(x, y)) = f(x, y) & x \in D, \\ \qquad\qquad\quad u(x, y) = 0 & x \in \partial D, \end{cases} \tag{4}$$

where a and f are random fields representing the diffusion coefficient and the forcing term. Typical UQ problems include evaluating the k-th moment, e.g. expectation, variance, of the solution field u or some functional $s(u)$ that depends on u, computing a failure probability of a system based on u, solving an optimization problem constrained by the PDE (3), calibrating the distribution of y, etc.

3. Verified dimension-adaptive hierarchical approximation

In this section, we present the dimension-adaptive hierarchical sparse grid approximation of high-dimensional UQ problems based on [16,21]. Our original contribution is to identify the stagnation phenomenon in the hierarchical construction of a generalized sparse grid and propose a verified 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.

3.1. Hierarchical interpolation and integration in one dimension

For numerical interpolation of continuous function $s \in C(\Gamma)$ in a one dimensional image $\Gamma \subset \mathbb{R}$, we first choose a set of collocation nodes (e.g. Clenshaw–Curtis nodes) ordered such that $y^1 < y^2 < \cdots < y^m \in \Gamma$ and for any given $y \in \Gamma$, we approximate the function value $s(y)$ by the interpolation formula

$$s(y) \approx \mathcal{U}s(y) = \sum_{j=1}^{m} s(y^j) l^j(y), \tag{5}$$

where $\mathcal{U}$ is an interpolation operator; l^j, $1 \leq j \leq m$ are basis functions that, depending on the regularity of the function s with respect to y in Γ, could be piecewise polynomials or global polynomials [29]. 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 $m(1) = 1$, and $m(i) = 2^{i-1} + 1$, $i \geq 1$. We consider nested set of nodes, i.e. $\Theta^i \subset \Theta^{i+1}$, $i = 1, 2, \dots, q$ with $q \in \mathbb{N}_+$. In this way, the hierarchical interpolation formula can be written as [5]

$$s(y) \approx \mathcal{U}^q s(y) = \sum_{i=1}^{q} \triangle^i s(y), \text{ where } \triangle^i = \mathcal{U}^i - \mathcal{U}^{i-1},\ 1 \leq i \leq q, \text{ and } \mathcal{U}^0 = 0. \tag{6}$$

For notational convenience, let us define $\Theta^i_\triangle = \Theta^i \setminus \Theta^{i-1}$, $1 \leq i \leq q$ with $\Theta^{i-1} := \emptyset$, and reorder the collocation nodes $y^1, \dots, y^{m(q)}$ in $\Theta^q = \cup_{i=1}^{q} \Theta^i_\triangle$ level by level in such a way that $y^i_j \in \Theta^i_\triangle$, $1 \leq i \leq q$, $1 \leq j \leq m(i) - m(i-1)$ with $m(0) = 0$. Corresponding to the reordering of the collocation nodes, we denote the basis functions as l^i_j, $1 \leq i \leq q$, $1 \leq j \leq m(i) - m(i-1)$. 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 (6) can be rewritten as

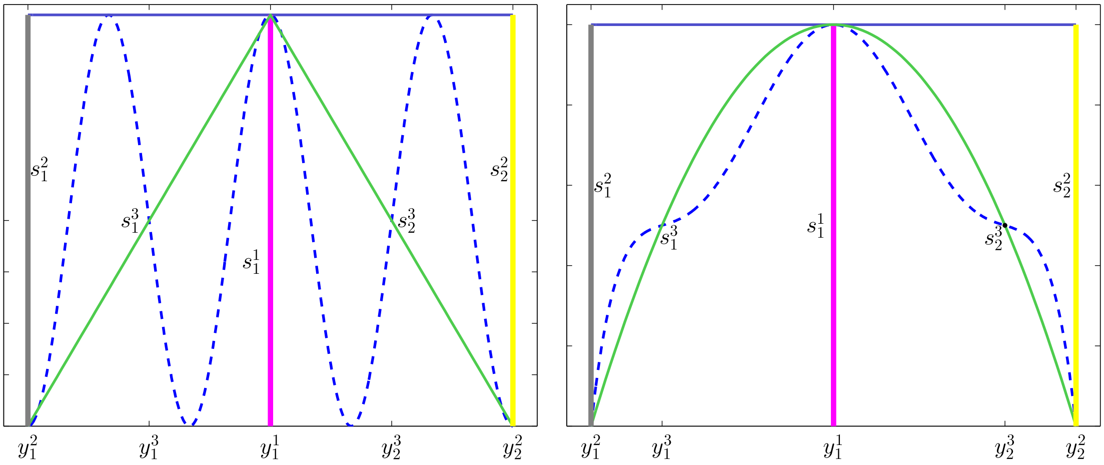

Fig. 1. Stagnation phenomenons for hierarchical interpolation. Left: piecewise linear polynomials based on equidistant nodes; right: Lagrange polynomials based on Clenshaw–Curtis nodes.

$$\mathcal{U}^q s(y) = \sum_{i=1}^{q} \left(\mathcal{U}^i s(y) - \mathcal{U}^i \circ \mathcal{U}^{i-1} s(y)\right) = \sum_{i=1}^{q} \sum_{y_j^i \in \Theta_\triangle^i} \underbrace{(s(y_j^i) - \mathcal{U}^{i-1} s(y_j^i))}_{s_j^i} l_j^i(y), \tag{7}$$

where real number s_j^i is called *hierarchical surplus* [5], 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.

As for numerical integration in evaluating statistical moments, we can take advantage of the interpolation formula (7) and assess the accuracy of integration by hierarchical surplus. For instance, the expectation of the function s can be computed by

$$\mathbb{E}[s] \approx \mathbb{E}[\mathcal{U}^q s] = \sum_{i=1}^{q} \sum_{y_j^i \in \Theta_\triangle^i} s_j^i w_j^i, \tag{8}$$

where the quadrature weights w_j^i are computed by $w_j^i = \int_\Gamma l_j^i(y)\rho(y)dy$ using suitable quadrature rules depending on the choice of collocation nodes [29]. Similarly, the k-th ($k \geq 2$) moment can be evaluated by setting the hierarchical surpluses as $s_j^i = s^k(y_j^i) - \mathcal{U}^{i-1} s^k(y_j^i)$, $1 \leq i \leq q$, $1 \leq j \leq m(i) - m(i-1)$. Based on the hierarchical surplus s_j^i, we may define the interpolation and integration errors $\mathcal{E}_i$, $\mathcal{E}_e$ as

$$\mathcal{E}_i := \max_{1 \leq j \leq m(q) - m(q-1)} |s_j^q|, \text{ and } \quad \mathcal{E}_e := \sum_{y_j^q \in \Theta_\triangle^q} s_j^q w_j^q. \tag{9}$$

There quantities can be used as error indicators in adaptively constructing the interpolation formula (7) and integration formula (8), 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 Fig. 1 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.

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

Algorithm 1 Verified hierarchical interpolation in one dimension.

```
 1: procedure INITIALIZATION:
 2:    specify error tolerance $\varepsilon_t$, type of interpolation basis functions $l(y)$ 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 (7);
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 (9);
      ..................................................................................................
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
```

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.

3.2. Dimension adaptive approximation for high-dimensional problems

In multiple dimensions, i.e. when $\Gamma \subset \mathbb{R}^K$, $K = 2, 3, \ldots$, the univariate interpolation formula (6) can be straightforwardly extended as the tensor product interpolation [1]

$$\mathcal{I}_q s(y) := \left(\mathcal{U}_1^q \otimes \cdots \otimes \mathcal{U}_K^q\right) s(y) = \sum_{i_1=1}^{q} \cdots \sum_{i_K=1}^{q} \left(\triangle_1^{i_1} \otimes \cdots \otimes \triangle_K^{i_K}\right) s(y), \tag{10}$$

where $\mathcal{U}_k^{q_k}$ and $\triangle_k^{i_k}$ are the univariate interpolation and difference operators in dimension $k = 1, \ldots, K$. As the tensor product interpolation needs too many collocation nodes, the Smolyak sparse grid interpolation [33] is employed to reduce the number of nodes, given by

$$\mathcal{S}_q s(y) = \sum_{|\mathbf{i}| \le q} \left(\triangle_1^{i_1} \otimes \cdots \otimes \triangle_K^{i_K}\right) s(y), \tag{11}$$

where the multivariate index $\mathbf{i} = (i_1, \ldots, i_K) \in \mathbb{N}_+^K$ $(N_+ = \{1, 2, \ldots, \})$ represents the 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 (11), we split it as follows

$$\mathcal{S}_q s(y) = \mathcal{S}_{q-1} s(y) + \triangle\mathcal{S}_q s(y), \text{ with } \triangle\mathcal{S}_q s(y) := \sum_{|\mathbf{i}| = q} \left(\triangle_1^{i_1} \otimes \cdots \otimes \triangle_K^{i_K}\right) s(y). \tag{12}$$

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

$$\triangle\mathcal{S}_q s(y) = \sum_{|\mathbf{i}|=q} \sum_{\mathbf{j} \in m_\triangle^{\mathbf{i}}} \underbrace{\left(s(y_{j_1}^{i_1}, \ldots, y_{j_K}^{i_K}) - \mathcal{S}_{q-1} s(y_{j_1}^{i_1}, \ldots, y_{j_K}^{i_K})\right)}_{s_{\mathbf{j}}^{\mathbf{i}}} \underbrace{\left(l_{j_1}^{i_1}(y_1) \otimes \cdots \otimes l_{j_K}^{i_K}(y_K)\right)}_{l_{\mathbf{j}}^{\mathbf{i}}}. \tag{13}$$

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; $m_\triangle^{\mathbf{i}}$ is the indicator set of the collocation nodes increased in the sparse grid due to the addition of the index $\mathbf{i}$.

The sparse grid interpolation is constructed in an isotropic manner due to the restriction $|\mathbf{i}| \le q$, which may still need too many nodes in high dimensional problems. For a more general construction of sparse grid interpolation that allows different interpolation levels in different dimensions, we break the isotropic restriction and pose only an *admissibility condition* to satisfy the essential property of the hierarchical representation [5,21]. 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 $k = 1, \dots, K$ 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* [5], 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 $S_t := \{\mathbf{i} \in \mathbb{N}_+^K : i_k \le q, 1 \le k \le K\}$. In the admissible index set S_m, being m the cardinality of S_m, we can write the generalized sparse grid interpolation in a hierarchical way as

$$\mathcal{S}_m s(y) = \sum_{\mathbf{i} \in S_m} \sum_{\mathbf{j} \in m_\triangle^{\mathbf{i}}} s_{\mathbf{j}}^{\mathbf{i}} l_{\mathbf{j}}^{\mathbf{i}}, \tag{14}$$

where $s_{\mathbf{j}}^{\mathbf{i}} = s(y_{\mathbf{j}}^{\mathbf{i}}) - \mathcal{S}_{m-1} s(y_{\mathbf{j}}^{\mathbf{i}})$. Correspondingly, the integration approximation is given by

$$\mathbb{E}[s] \approx \mathbb{E}[\mathcal{S}_m s] = \sum_{\mathbf{i} \in S_m} \sum_{\mathbf{j} \in m_\triangle^{\mathbf{i}}} s_{\mathbf{j}}^{\mathbf{i}} w_{\mathbf{j}}^{\mathbf{i}}, \tag{15}$$

where the weight $w_{\mathbf{j}}^{\mathbf{i}} = \int_\Gamma \left(l_{j_1}^{i_1}(y_1) \otimes \cdots \otimes l_{j_K}^{i_K}(y_K) \right) \rho(y) dy$ is computed by a suitable quadrature rule depending on the choice of nodes. We present the construction of the generalized sparse grid as follows: 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 [21] to use the averaged hierarchical surplus as the error indicator to pick $\mathbf{i}$

$$\mathbf{i} = \operatorname*{argmax}_{\mathbf{i} \in \mathscr{A}} \mathcal{E}_i(\mathbf{i}) \text{ with } \mathcal{E}_i(\mathbf{i}) := \frac{1}{n(\mathbf{i})} \sum_{\mathbf{j} \in m_\triangle^{\mathbf{i}}} |s_{\mathbf{j}}^{\mathbf{i}}|, \tag{16}$$

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

$$\mathcal{E}_e(\mathbf{i}) := \frac{1}{n(\mathbf{i})} \left| \sum_{\mathbf{j} \in m_\triangle^{\mathbf{i}}} s_{\mathbf{j}}^{\mathbf{i}} w_{\mathbf{j}}^{\mathbf{i}} \right|, \tag{17}$$

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 (17) tends to underestimate the integral error since only one index is considered. We provide a more reasonable estimate for the integral error as

$$\mathcal{E}_e(\mathscr{A}) = \left| \sum_{\mathbf{i} \in \mathscr{A}} \sum_{\mathbf{j} \in m_\triangle^{\mathbf{i}}} s_{\mathbf{j}}^{\mathbf{i}} w_{\mathbf{j}}^{\mathbf{i}} \right|. \tag{18}$$

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 univariate 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 [5,21] 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 $\mathbf{i} = \operatorname{argmin}_{\mathbf{k} \in \mathscr{A}} |\mathbf{k}|$ to the active index set $\mathscr{A}$ if [21]

$$\frac{\min_{\mathbf{i} \in \mathscr{A}} |\mathbf{i}|}{\max_{\mathbf{i} \in \mathscr{A} \cup \mathscr{O}} |\mathbf{i}|} \le (1 - w), \tag{19}$$

regardless of the current error indicator; where $w = 1$ corresponds to the purely greedy adaptive construction and $w = 0$ to 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

Algorithm 2 Verified dimension-adaptive hierarchical algorithm for interpolation.

```
 1: procedure INITIALIZATION:
 2:     specify error tolerance $\varepsilon_t$, types of interpolation basis functions $l(y)$ 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 $s_{\mathbf{j}}^{\mathbf{1}} = s(y_{\mathbf{j}}^{\mathbf{1}})$, $y_{\mathbf{j}}^{\mathbf{1}} \in \Theta^{\mathbf{1}}$;
 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 $m \le M$ 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 $y_{\mathbf{j}}^{\mathbf{i}} \in \Theta_\triangle$, evaluate function values $s(y_{\mathbf{j}}^{\mathbf{i}})$ and the interpolation $\mathcal{S}_m s(y_{\mathbf{j}}^{\mathbf{i}})$ by (14);
11:         compute the hierarchical surpluses $s_{\mathbf{j}}^{\mathbf{i}} = s(y_{\mathbf{j}}^{\mathbf{i}}) - \mathcal{S}_m s(y_{\mathbf{j}}^{\mathbf{i}})$ and error indicator $\mathcal{E}_i$ by (16);
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 $\mathcal{E}_i(\mathbf{i}_v) \le \varepsilon_t$ 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 (16) 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 $\mathbf{i} = \mathrm{argmax}_{\mathbf{i} \in \mathscr{A}} \mathcal{E}_i(\mathbf{i})$;
27:         if $\mathcal{E}_i(\mathbf{i}) \le \varepsilon_t$ then
28:             return .
29:         end if
30:     end while
31: end procedure
```

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 (16) by the integration error indicator $\mathcal{E}_e$ in (17).

As pointed out in [14,22], 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 computationally expensive, especially for high-dimensional UQ problems with verification procedure, where costly solution of a large number of PDE is required. Fortunately, this computational burden can be considerably alleviated by using the adaptive reduced basis method that will be developed in the next section, where full solve of the underlying PDE model is replaced by a very cheap solve of a reduced model.

4. Adaptive and weighted reduced basis method

In this section, we develop an adaptive and weighted reduced basis method in combination with the dimension-adaptive sparse grid approximation to efficiently solve high-dimensional UQ problems.

4.1. Reduced basis approximation

We take the typical problem (4) for the presentation of the reduced basis approximation and refer to [30,28] and the references therein for more general problems. The semi-weak formulation of (4) can be written as: given $y \in \Gamma$, find $u(y) \in H_0^1(D)$ such that

$$A(u, v; y) = F(v; y) \quad \forall v \in H_0^1(D), \tag{20}$$

where $H_0^1(D)$ is the classical Hilbert space for functions with null boundary trace; A and F are the y-dependent bilinear and linear forms corresponding to the diffusion and the forcing terms in (4).

Let $X \subset H_0^1(D)$ denote a finite-dimensional subspace for the approximation of the PDE solution in physical space, for instance the high-fidelity finite element approximation space [26]. Then the reduced basis problem is formulated as: find $u_N \in X_N$ such that

$$A(u_N, v_N; y) = F(v_N; y) \quad \forall v_N \in X_N, \tag{21}$$

Algorithm 3 Adaptive greedy algorithm based on hierarchical sparse grid approximation.

```
1: procedure INITIALIZATION:
2:     specify error tolerance $\epsilon_t$, solve (20) 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 (21), 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 (20) 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
```

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

$$X_N = \text{span}\{u(y^n), 1 \leq n \leq N\}. \tag{22}$$

In order to guarantee algebraic stability in solving (21), we perform Gram–Schmidt orthogonalization procedure [30] on the snapshots $u(y^1), \dots, u(y^N)$ and obtain a set of orthonormal basis functions $\zeta_1, \dots, \zeta_N$ 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 small as possible, so that problem (21) can be very cheap to solve.

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 $y \in \Gamma$ [30], i.e.

$$y^{N+1} = \underset{y \in \Gamma}{\text{argmax}}\, \mathcal{E}_r(y), \text{ with } \mathcal{E}_r(y) := ||u(y) - u_N(y)||_X. \tag{23}$$

However, to solve problem (23) is computationally unfeasible since a full solve of a high-fidelity problem is needed to evaluate $u(y)$ at $y \in \Gamma$. To deal with this difficulty, we first propose an adaptive greedy algorithm based on the hierarchical sparse grid approximation, which replaces the infinite set Γ by a finite set of collocation nodes in Algorithm 3, and then use the weighted a posteriori error bound as presented in the next section in order to avoid the full solve of the high-fidelity problem.

The adaptive greedy Algorithm 3 for the construction of the reduced basis space explores all the collocation nodes in the construction of the dimension-adaptive hierarchical sparse grid approximation in lines 5 and 6 of Algorithm 3. As a result, the quantity of interest s is evaluated based on the surrogate (reduced basis) computationally cheap solution of the reduced basis problem rather than on the solution of the expensive high-fidelity problem. The computational cost is therefore dramatically reduced. Moreover, error estimates of the surrogate outputs of interest can be obtained based on the reduced basis approximation error $\mathcal{E}_r$, which is 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 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 [30]: for every $y \in \Gamma$, let $R(v; y) \in X'$ be the residual in the dual space of X, defined as

$$R(v; y) := F(v; y) - A(u_N(y), v; y) \quad \forall v \in X. \tag{24}$$

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

$$(\hat{e}(y), v)_X = R(v; y) \quad \forall v \in X, \tag{25}$$

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 (20), (21) and (24)

$$A(e(y), v; y) = R(v; y) \quad \forall v \in X. \tag{26}$$

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

$$\alpha(y)||e(y)||_X^2 \leq A(e(y), e(y); y) = R(e(y); y) \leq ||R(\cdot, y)||_{X'}||e(y)||_X = ||\hat{e}(y)||_X||e(y)||_X, \tag{27}$$

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

$$\triangle_N^u(y) := ||\hat{e}(y)||_X/\alpha(y), \tag{28}$$

yielding $||u(y) - u_N(y)||_X \leq \triangle_N^u(y)$ by (27). 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

$$|s(y) - s_N(y)| = |F(u(y); y) - F(u_N(y); y)| = A(e(y), e(y); y) \leq ||\hat{e}(y)||_X^2/\alpha(y) =: \triangle_N^s(y). \tag{29}$$

We remark that the error bounds (28) and (29) 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 (21) 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 (20) 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 [7] as

$$\triangle_N^{\rho,u}(y) = \sqrt{\rho(y)}\triangle_N^u(y), \text{ or } \triangle_N^{\rho,s}(y) = \rho(y)\triangle_N^s(y), \tag{30}$$

which put small weight on the samples with small probability density, thus generating relatively less basis functions 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 [7]. In summary, computing the error indicators (30) 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 [20] 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.

4.4. Offline–online decomposition

In the construction of the reduced basis space with a small number of basis functions 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 based on the assumption that the random fields admit an affine decomposition

$$a(x, y) = \sum_{q=1}^{Q_a} \Theta_q^a(y)a_q(x) \text{ and } f(x, y) = \sum_{q=1}^{Q_f} \Theta_q^f(y)f_q(x), \tag{31}$$

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 [31]. Moreover, for any nonaffine random fields, we may apply empirical interpolation [3] to obtain an affine approximation as in (31). Under such assumption, we have

$$A(u, v; y) = \sum_{q=1}^{Q_a} \Theta_q^a(y)A_q(u, v) \text{ and } F(v; y) = \sum_{q=1}^{Q_f} \Theta_q^f(y)F_q(v), \tag{32}$$

with the definition $A_q(u, v) := (a_q\nabla u, \nabla v)$, $1 \leq q \leq Q_a$ and $F_q(v) := (f_q, v)$, $1 \leq q \leq Q_f$.

The reduced basis solution $u_N(y)$ projected on the reduced basis functions $\zeta_1, \ldots, \zeta_N$ is written as

$$u_N(y) = \sum_{m=1}^{N} u_{Nm}(y)\zeta_m, \tag{33}$$

which is substituted in the Galerkin projection problem (21) as: find $u_{Nm}(y)$, $1 \leq m \leq N$ such that

$$\sum_{m=1}^{N}\sum_{q=1}^{Q_a} \Theta_q^a(y)A_q(\zeta_m, \zeta_n)u_{Nm}(y) = \sum_{q=1}^{Q_f} \Theta_q^f(y)F_q(\zeta_n), \quad 1 \leq n \leq N. \tag{34}$$

Here, the matrix $A_q(\zeta_m, \zeta_n)$, $1 \leq q \leq Q_a$, $1 \leq m, n \leq N$ and the vector $F_q(\zeta_n)$, $1 \leq q \leq Q_f$, $1 \leq n \leq N$ 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 (34) 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

$$s_N(y) = F(u_N(y); y) = \sum_{n=1}^{N} \left(\sum_{q=1}^{Q_f} \Theta_q^f(y) F_q(\zeta_n) \right) u_{Nn}(y). \tag{35}$$

As for the evaluation of $||\hat{e}(y)||_X^2$ in (29), we first expand the residual (24) as

$$R(v; y) = F(v; y) - A(u_N, v; y) = \sum_{q=1}^{Q_f} \Theta_q^f(y) F_q(v) - \sum_{n=1}^{N} \left(\sum_{q=1}^{Q_a} \Theta_q^a(y) A_q(\zeta_n, v) \right) u_{Nn}(y). \tag{36}$$

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^n (defined as $A_q^n(v) = -A_q(\zeta_n, v)$, $\forall v \in X$) in X. By recalling (25) we have

$$\hat{e}(y) = \sum_{q=1}^{Q_f} \Theta_q^f(y) \mathcal{C}_q - \sum_{n=1}^{N} \left(\sum_{q=1}^{Q_a} \Theta_q^a(y) \mathcal{L}_q^n \right) u_{Nn}(y), \tag{37}$$

so that

$$\begin{aligned} ||\hat{e}(y)||_X^2 = & \sum_{q}^{Q_f} \sum_{q'}^{Q_f} \Theta_q^f(y) \Theta_{q'}^f(y) (\mathcal{C}_q, \mathcal{C}_{q'})_X \\ & + 2 \sum_{n=1}^{N} \sum_{q=1}^{Q_f} \sum_{q'=1}^{Q_a} \Theta_q^f(y) \Theta_{q'}^a(y) (\mathcal{C}_q, \mathcal{L}_{q'}^n)_X u_{Nn}(y) \\ & + \sum_{n=1}^{N} \sum_{n'=1}^{N} \sum_{q=1}^{Q_a} \sum_{q'=1}^{Q_a} \Theta_q^a(y) \Theta_{q'}^a(y) u_{Nn}(y) (\mathcal{L}_q^n, \mathcal{L}_{q'}^{n'})_X u_{Nn'}(y). \end{aligned} \tag{38}$$

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 (38) with $O(Q_f^2 + N Q_f Q_a + N^2 Q_a^2)$ operations.

Note that when the number of terms Q_f and Q_a become large, the full online evaluation of (38) is 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 $(1, 0.5, 0.5, \dots, 0.5)$ is a neighbor of the node $(0, 0.5, 0.5, \dots, 0.5)$, which are only different in the first dimension. Based on these two observations, we may identify the different terms in (38) 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 $O(Q_f + N Q_a)$ operations in average for each evaluation. We remark that this computational reduction is still valid whenever there are only a few terms among $\Theta_q^a(y)$, $1 \le q \le Q_a$ different from one node to its neighbors. However, when this circumstance does not occur, the evaluation of (38) remains costly and involves $O(Q_f^2 + N Q_f Q_a + N^2 Q_a^2)$ operations.

5. Numerical experiments

This section is devoted to demonstrate the efficiency and accuracy of the adaptive and reduced computational algorithm. We illustrate the computational performance of the proposed Algorithm 2 with verification in two dimensions, and compare it to the algorithm without verification. Sections 5.2 and 5.3 deal with two applications and demonstrate how the proposed algorithm can be effectively applied to reduce the computational effort. In both cases, the solution manifold is low dimensional.

5.1. Hierarchical construction with verification

In the first experiment, we compare the verified dimension-adaptive hierarchical interpolation Algorithm 2 with the same algorithm but without verification. We consider $s : [0, 1]^2 \to \mathbb{R}$, given by

$$s(y) = \cos(2\pi(y_1 - 0.3)) \cos(2\pi(y_2 - 0.5)). \tag{39}$$

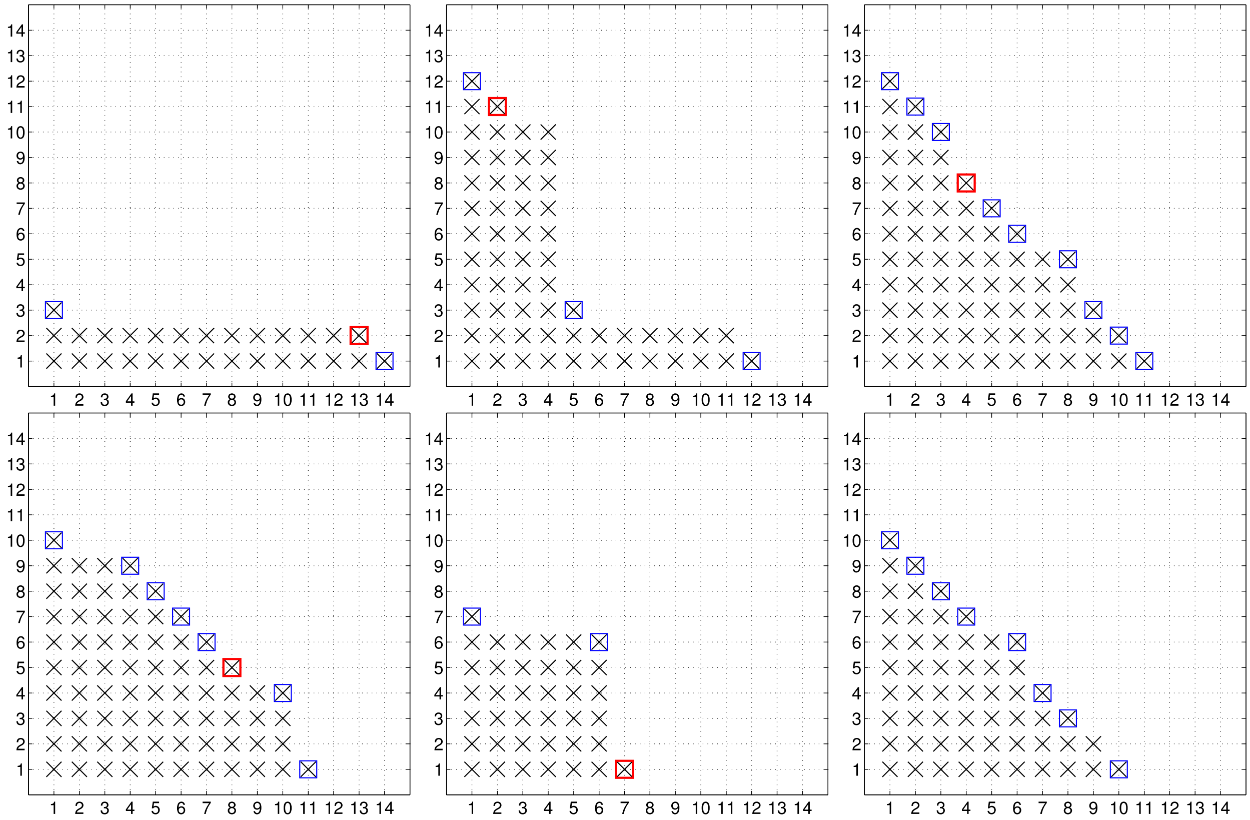

Fig. 2. Illustration of dimension-adaptive hierarchical construction of the generalized sparse grid in different cases; top row: piecewise interpolation without verification with weight $w = 1$ (left), $w = 0.5$ (middle), and $w = 0$ (right); bottom row: piecewise interpolation with verification and weight $w = 1$ (left), Lagrange interpolation with verification and weight $w = 1$ (middle), and $w = 0$ (right). (For interpretation of the references to colour in this figure, the reader is referred to the web version of this article.)

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 (19) 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 $\max_{y \in \Xi_{test}} |s(y) - \mathcal{S}_m s(y)|$ with the set of testing nodes given by $\Xi_{test} := \{y_1, y_2 = n/2^8, n = 0, \dots, 2^8\}$, a fine regular grid with step size $1/2^8$. The constructed index sets S_m for the six different cases are plotted in Fig. 2, 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. Fig. 3 reports the interpolation errors.

From the first figure (left-top of Fig. 2), 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 Fig. 3) since the function is not sufficiently well approximated in the second dimension. The balancing scheme with $w = 0.5$ (see middle-top of Fig. 2) 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 Fig. 3 (left). The Smolyak isotropic sparse grid construction does not run into the stagnation problem and achieves small interpolation error in this example (see right-top of Fig. 2), 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 Fig. 2) and the sparse scheme that produces many more useless nodes in each single dimension (see right-bottom of Fig. 2). Note that the last two cases result in higher approximation accuracy (see Fig. 3) 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 Fig. 2 (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

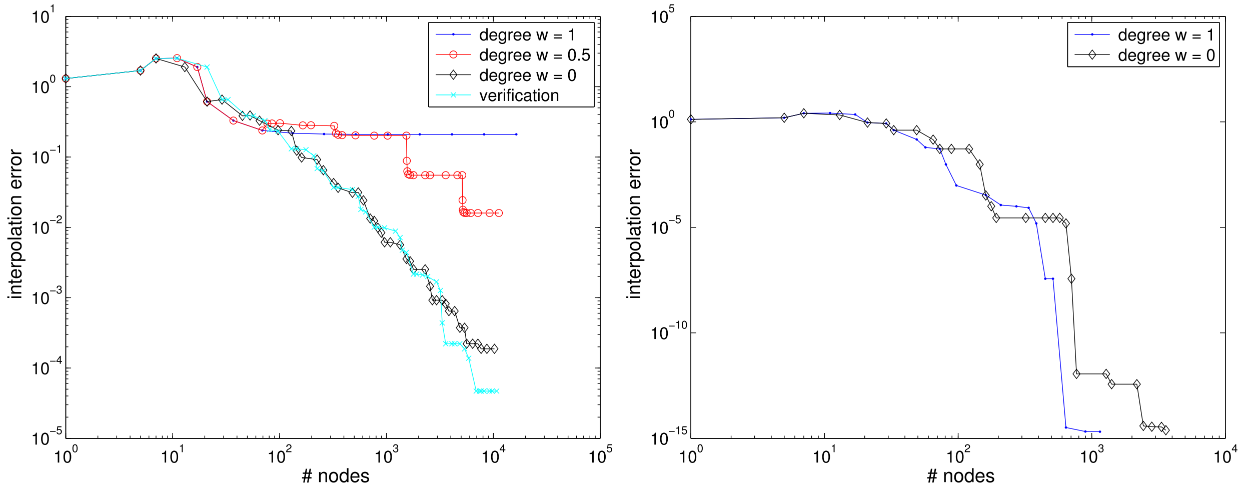

Fig. 3. Interpolation error corresponding to the grid construction in Fig. 2; left: piecewise interpolation in the first four cases; right: Lagrange interpolation in the last two cases.

is to blame as one drawback of the dimension-adaptive hierarchical construction approach. We remark that the balancing scheme in (19) cannot 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.

5.2. Heat diffusion in thermal blocks

In this example, we study a heat diffusion problem (4) 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, $1 \leq k \leq K$. The thermal conductivity of each block is a random variable. In the first test, we demonstrate the efficiency of the weighted a posteriori error bound (30) in the case of arbitrary probability measure for integration problem. We consider the random coefficient a in (4) as

$$a(x, y) = \sum_{k=1}^{K} \chi_{D_k}(x) 10^{(y_k - 0.5)}, \tag{40}$$

where χ_{D_k} is a characteristic function supported on the block D_k and $y_k \in [0, 1]$, $1 \leq k \leq K$ 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 (17) is used with the total number of nodes specified as 10^n, $0 \leq n \leq 4$ 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 (29). We apply both the a posteriori error bound (28) and the weighted a posteriori error bound (30) to construct the reduced basis space, resulting in 211 and 118 basis functions, respectively.

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 Fig. 4. From the right of this figure we can see that the reduced basis approximation with the weighted a posteriori error bound (30) achieves almost the same accuracy for integration as that without the weight (28), even using much less basis functions (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 basis functions, see in the left of Fig. 4. 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 (28) and (30).

Fig. 5 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 basis functions. In fact, the probability density ρ in (30) 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 basis functions. Moreover, this test also demonstrates that the total number of reduced basis functions is much smaller than the total number of constructed nodes, thus efficiently alleviate the entire computational cost.

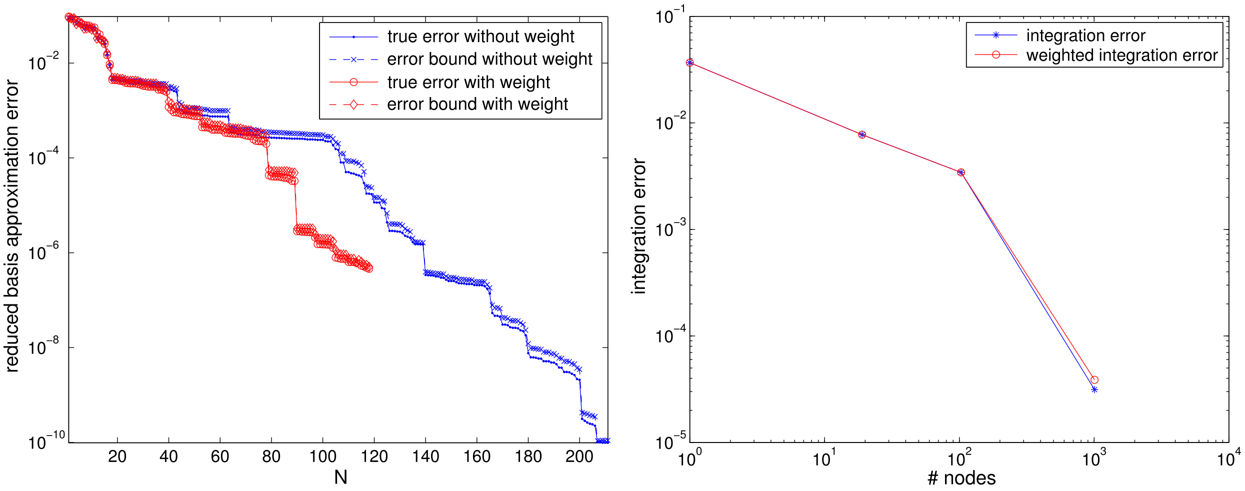

Fig. 4. Left: true error and error bound of reduced basis approximation constructed by a posteriori error bound without (28) and with weight (30); right: (weighted) integration error.

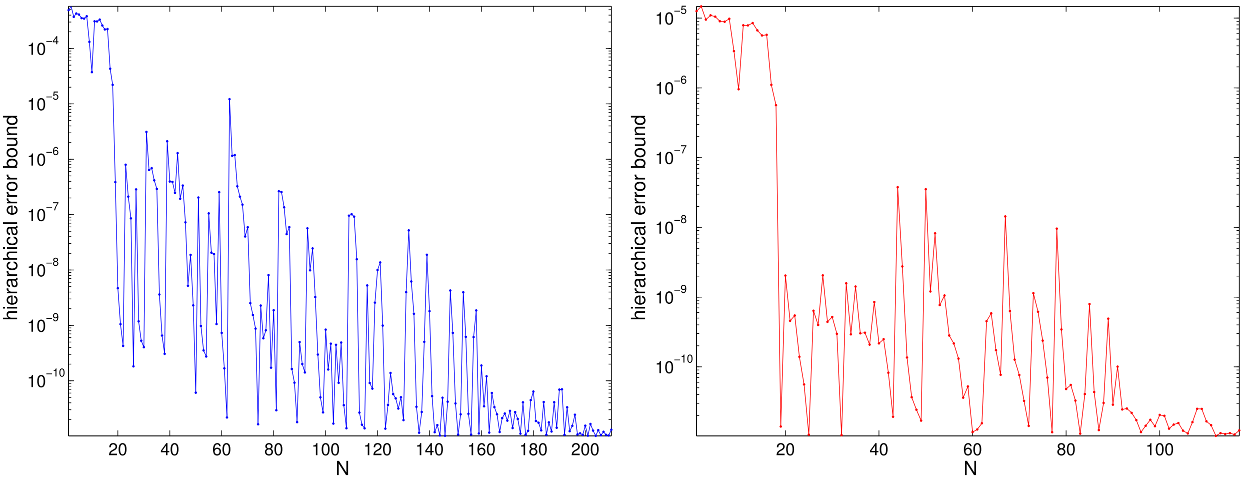

Fig. 5. Left: the a posteriori error bound (28); right: the weighted a posteriori error bound (30) during the hierarchical construction of the generalized sparse grid by Algorithm 2.

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

$$a(x, y) = \sum_{k=1}^{K} \chi_{D_k}(x) 10^{c_k(y_k - 0.5)} \tag{41}$$

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 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 basis functions 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.

Fig. 6 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 basis functions 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

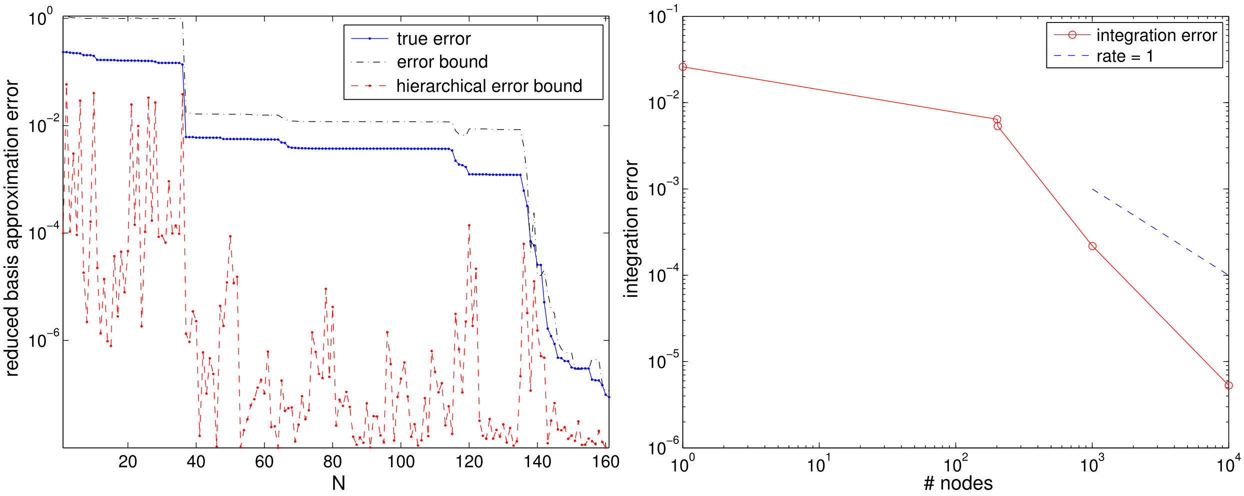

Fig. 6. Left: true error and error bound of reduced basis approximation; right: integration error.

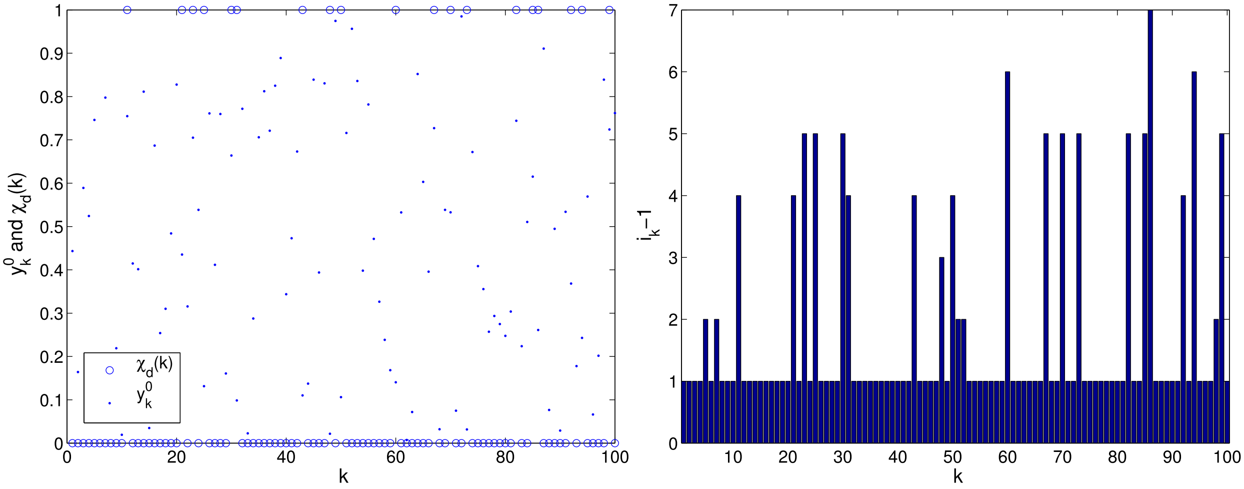

Fig. 7. Left: true error and error bound of reduced basis approximation; right: integration error.

to the error tolerance and remain smaller than the maximum error bounds at the 100 samples. Fig. 7 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 Fig. 7, where a larger value of y_k^0 leads to a relatively deeper grid level.

5.3. 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

$$\begin{cases} -\nabla(a\nabla p) = 0 & \text{in } D, \\ p = 1 & \text{on } \partial D_4, \\ p = 0 & \text{on } \partial D_2, \\ a\nabla p \cdot \mathbf{n} = 0 & \text{on } \partial D_1 \cup \partial D_3, \end{cases} \tag{42}$$

where the physical domain is the two dimensional square $D = (0, 1)^2$, as shown in Fig. 8, 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 $x = (x_1, x_2)$)

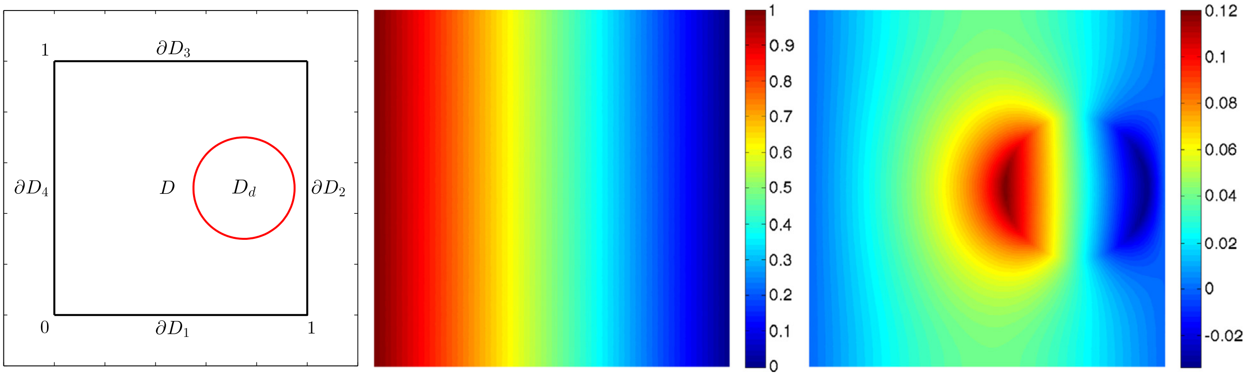

Fig. 8. Left: physical domain and boundaries; middle and right: primal and dual solutions. (For interpretation of the colours in this figure, the reader is referred to the web version of this article.)

$$a(x,y) = \mathbb{E}[a] + \left(\frac{\sqrt{\pi}L}{2}\right)^{1/2} y_1 + \sum_{k=1}^{K} \sqrt{\lambda_k}\left(\sin(k\pi x_1)y_{2k} + \cos(k\pi x_1)y_{2k+1}\right), \tag{43}$$

which is a truncated Karhunen–Loève expansion of a Gauss covariance kernel $\exp(-(x_1 - x_1')^2/L^2)$ with correlation length L [24]. The eigenvalues λ_k, $1 \le k \le K$, of this kernel decay exponentially as

$$\lambda_k = \sqrt{\pi}L\exp\left(-\frac{(k\pi L)^2}{4}\right), \tag{44}$$

and the random variables y_k, $1 \le k \le 2K+1$, 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

$$s(y) := L(p; y) = \int_{D_d} a(x,y)\partial_{x_1} p(x,y)dx, \tag{45}$$

where the disk region D_d has center $(0.75, 0.5)$ and radius 0.2, see Fig. 8. This quantity is not compliant with the right hand side of equation $(42)_1$. Therefore, we adopt a primal-dual approach [30]. We first write the weak formulation of the Darcy equation (42) as: find $p \in H^1(D)$ such that

$$A(p,q;y) = 0 \quad \forall q \in H^1_{dir}(D), \tag{46}$$

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

$$A = \sum_{k=0}^{2K+1} A_k(p,q)y_k, \tag{47}$$

being A_k defined corresponding to the terms in the expansion of the permeability coefficient a in (43) and $y_0 = 1$ for notational convenience. The dual problem associated with the primal problem (46) for the quantity of interest s is formulated as: find $\varphi \in H^1_{dir}$ such that

$$A(q,\varphi;y) = -L(q;y) \quad \forall q \in H^1_{dir}(D). \tag{48}$$

We construct reduced basis space $X^{pr}_{N_{pr}}$ with N_{pr} basis functions and $X^{du}_{N_{du}}$ with N_{du} basis functions to approximate the primal and dual weak problems (46) and (48) and define the residual of each problem as

$$R^{pr}(q;y) = -A(p_{N_{pr}}, q; y) \text{ and } R^{du}(q;y) = -L(q;y) - A(q, \varphi_{N_{du}}; y), \tag{49}$$

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 (45) by

$$s_N(y) = L(p_{N_{pr}}(y); y) - R^{pr}(\varphi_{N_{du}}(y); y), \tag{50}$$

whose error can be bounded as (see details in [30])

$$|s(y) - s_N(y)| \le \triangle^s_N(y) := \frac{||R^{pr}(\cdot;y)||_{X'}||R^{du}(\cdot;y)||_{X'}}{\alpha(y)}. \tag{51}$$

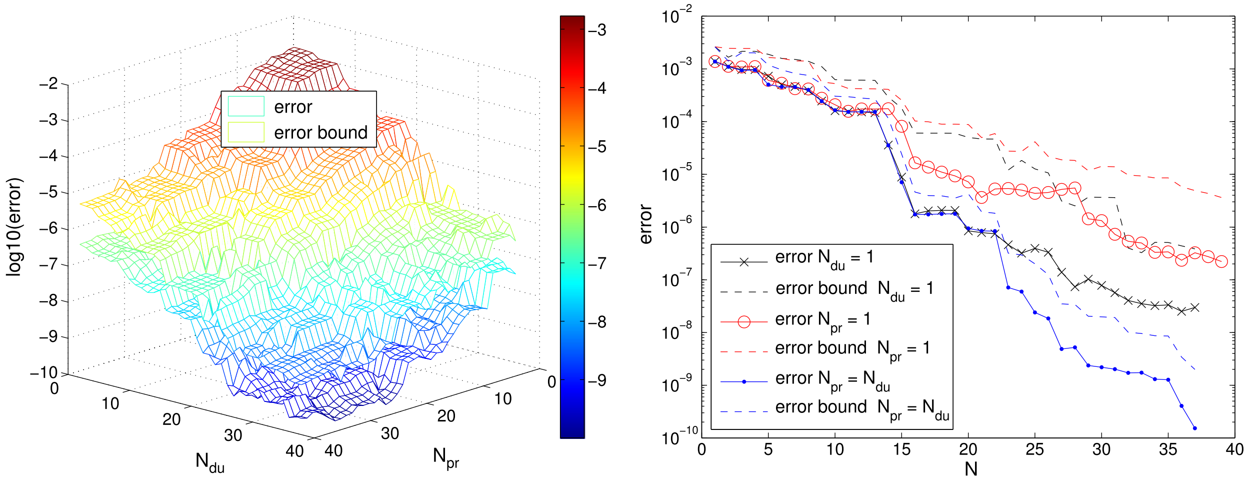

Fig. 9. Left: reduced basis approximation error and error bound w.r.t. the number of primal basis functions N_{pr} and the number of dual basis functions N_{du}; right: three different settings of N_{pr} and N_{du}, $K = 8$. (For interpretation of the colours in this figure, the reader is referred to the web version of this article.)

For the approximation in physical space, we use piecewise linear finite element basis on a regular triangular mesh with 17 361 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 (17) 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 (51) 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 (44) and K as $8, 16, 32, 64$, which lead to $17, 33, 65, 129$ dimensions, resulting in approximation of the coefficient given in (43) with accuracy 59%, 89%, 99% and 100%. A set of typical solutions of the primal problem (46) and dual problem (48) at a randomly selected sample are depicted in Fig. 8 (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 (50).

In the 17 dimensional case ($K = 8$), 37 primal basis functions and 39 dual basis functions are constructed. We test the convergence of the worst reduced basis approximation error with respect to the number of primal basis functions and dual basis functions computed with 100 randomly selected samples, which is displayed in Fig. 9. From the left figure, we can observe that both the approximation error and the error bound decrease with growing number of primal and dual basis functions, 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 (51). The interpolation errors by the hierarchical interpolation formula (14) 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 algorithm. 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 to $17, 33, 65, 129$ dimensional problems, are reported in Fig. 10. The number of primal and dual basis functions 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 that only a few tens (about 50) of reduced basis functions 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 otherwise. This result is in line with the theoretical prediction of [10]. When the random input features ℓ_p sparsity, so does the solution and related quantities of interest. Consequently, the sparse approximation converges with a rate independent of the dimensions, see [10] for more details. Furthermore, as shown in the right of Fig. 10, the integration error converges with rate larger than 1, which demonstrates that our computational strategy for integration in high dimensions is very efficient.

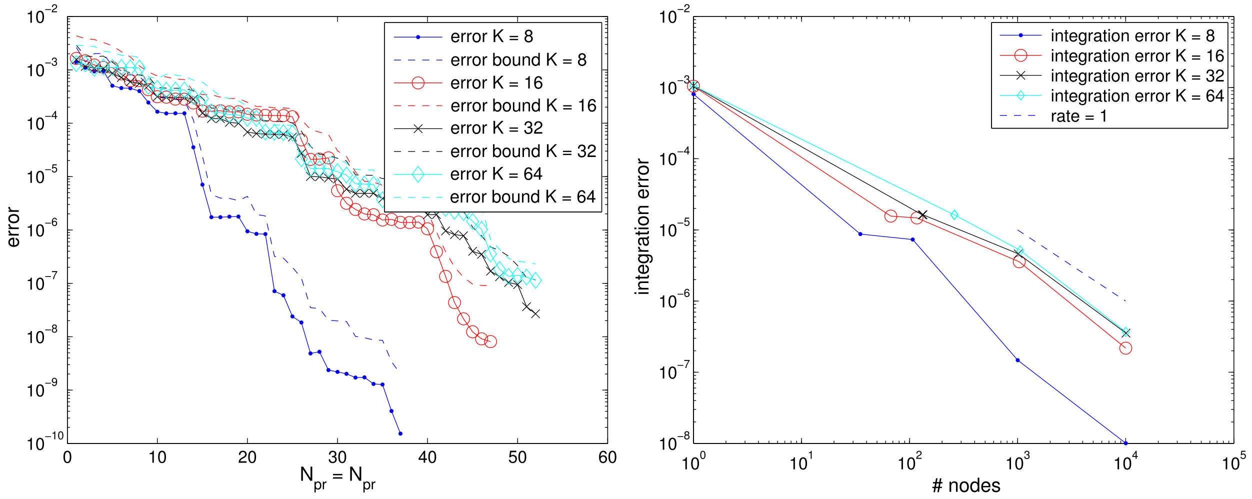

Fig. 10. Left: reduced basis approximation error and error bound with $N_{pr} = N_{du}$; right: integration error. $K = 8, 16, 32, 64$, corresponding to 17, 33, 65, 129 dimensional problems.

6. Concluding remarks

In this paper we have developed an adaptive reduced computational algorithm for solving high-dimensional 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 depending 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 basis functions without taking into account the underlying PDE models. Furthermore, only a few basis functions, 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 dramatically alleviates the prohibitive computational effort in solving large-scale PDE models.

Several further topics are to be investigated in applying the adaptive and reduced computational algorithm for more general 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, e.g. by checking the pointwise hierarchical surpluses instead of an averaged or maximum value at one index [22]. 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 basis functions but project the new solution into the reduced basis space spanned by solutions at some selected samples [6]. Another research topic is to develop more specific and goal-oriented reduced basis method 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 can be naturally combined with 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 demonstrated in [6] for risk analysis.

Acknowledgements

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, R. Tempone, A stochastic collocation method for elliptic partial differential equations with random input data, SIAM J. Numer. Anal. 45 (3) (2007) 1005–1034.

[2] I. Babuška, R. Tempone, G.E. Zouraris, Galerkin finite element approximations of stochastic elliptic partial differential equations, SIAM J. Numer. Anal. 42 (2) (2005) 800–825.

[3] M. Barrault, Y. Maday, N.C. Nguyen, A.T. Patera, An empirical interpolation method: application to efficient reduced-basis discretization of partial differential equations, C. R. Math. Anal. Numér. 339 (9) (2004) 667–672.

[4] S. Boyaval, C. Le Bris, T. Lelièvre, Y. Maday, N.C. Nguyen, A.T. Patera, Reduced basis techniques for stochastic problems, Arch. Comput. Methods Eng. 17 (2010) 435–454.

[5] H.J. Bungartz, M. Griebel, Sparse grids, Acta Numer. 13 (1) (2004) 147–269.

[6] P. Chen, A. Quarteroni, Accurate and efficient evaluation of failure probability for partial differential equations with random input data, Comput. Methods Appl. Mech. Eng. 267 (2013) 233–260.

[7] P. Chen, A. Quarteroni, G. Rozza, A weighted reduced basis method for elliptic partial differential equations with random input data, SIAM J. Numer. Anal. 51 (6) (2013) 3163–3185.

[8] P. Chen, A. Quarteroni, G. Rozza, Comparison of reduced basis and stochastic collocation methods for elliptic problems, J. Sci. Comput. 59 (2014) 187–216.

[9] A. Chkifa, A. Cohen, C. Schwab, Breaking the curse of dimensionality in sparse polynomial approximation of parametric PDEs, J. Math. Pures Appl. (2014).

[10] A. Cohen, R. Devore, C. Schwab, Analytic regularity and polynomial approximation of parametric and stochastic elliptic PDE's, Anal. Appl. 9 (01) (2011) 11–47.

[11] J. Dick, F.Y. Kuo, I.H. Sloan, High-dimensional integration – the Quasi-Monte Carlo way, Acta Numer. 22 (2013) 133–288.

[12] H.C. Elman, Q. Liao, Reduced basis collocation methods for partial differential equations with random coefficients, SIAM/ASA J. Uncertain. Quantificat. 1 (1) (2013) 192–217.

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

[14] J. Foo, G.E. Karniadakis, Multi-element probabilistic collocation method in high dimensions, J. Comput. Phys. 229 (5) (2010) 1536–1557.

[15] Z. Gao, J.S. Hesthaven, On ANOVA expansions and strategies for choosing the anchor point, Appl. Math. Comput. 217 (7) (2010) 3274–3285.

[16] T. Gerstner, M. Griebel, Dimension-adaptive tensor-product quadrature, Computing 71 (1) (2003) 65–87.

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

[18] B. Haasdonk, K. Urban, B. Wieland, Reduced basis methods for parameterized partial differential equations with stochastic influences using the Karhunen–Loève expansion, SIAM/ASA J. Uncertain. Quantificat. 1 (1) (2013) 79–105.

[19] J.S. Hesthaven, 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.

[20] D.B.P. Huynh, G. Rozza, S. Sen, A.T. Patera, A successive constraint linear optimization method for lower bounds of parametric coercivity and inf-sup stability constants, C. R. Math. Anal. Numér. 345 (8) (2007) 473–478.

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

[22] X. Ma, N. Zabaras, An adaptive high-dimensional stochastic model representation technique for the solution of stochastic partial differential equations, J. Comput. Phys. 229 (10) (2010) 3884–3915.

[23] F. Nobile, R. Tempone, C.G. Webster, An anisotropic sparse grid stochastic collocation method for partial differential equations with random input data, SIAM J. Numer. Anal. 46 (5) (2008) 2411–2442.

[24] F. Nobile, R. Tempone, C.G. Webster, A sparse grid stochastic collocation method for partial differential equations with random input data, SIAM J. Numer. Anal. 46 (5) (2008) 2309–2345.

[25] A. Nouy, Proper generalized decompositions and separated representations for the numerical solution of high dimensional stochastic problems, Arch. Comput. Methods Eng. 17 (4) (2010) 403–434.

[26] A. Quarteroni, Numerical Models for Differential Problems, MS & A, vol. 8, Springer, 2013.

[27] A. Quarteroni, A. Manzoni, F. Negri, Reduced Basis Methods for Partial Differential Equations. An Introduction, UNITEXT Series, Springer, 2015, in press.

[28] A. Quarteroni, G. Rozza, A. Manzoni, Certified reduced basis approximation for parametrized partial differential equations and applications, J. Math. Ind. 1 (1) (2011) 1–49.

[29] A. Quarteroni, R. Sacco, F. Saleri, Numerical Mathematics, Springer, 2007.

[30] G. Rozza, D.B.P. Huynh, A.T. Patera, Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations, Arch. Comput. Methods Eng. 15 (3) (2008) 229–275.

[31] C. Schwab, R.A. Todor, Karhunen–Loève approximation of random fields by generalized fast multipole methods, J. Comput. Phys. 217 (1) (2006) 100–122.

[32] C. Schwab, R.A. Todor, Sparse finite elements for elliptic problems with stochastic loading, Numer. Math. 95 (4) (2003) 707–734.

[33] S.A. Smolyak, Quadrature and interpolation formulas for tensor products of certain classes of functions, Dokl. Akad. Nauk SSSR 4 (1963) 240–243.

[34] D. Xiu, J.S. Hesthaven, High-order collocation methods for differential equations with random inputs, SIAM J. Sci. Comput. 27 (3) (2005) 1118–1139.

[35] D. Xiu, G.E. Karniadakis, The Wiener–Askey polynomial chaos for stochastic differential equations, SIAM J. Sci. Comput. 24 (2) (2003) 619–644.