Multilevel tensor approximation of PDEs with random data

J. Ballani, D. Kressner, M. Peters

MULTILEVEL TENSOR APPROXIMATION OF PDES WITH RANDOM DATA

JONAS BALLANI, DANIEL KRESSNER, AND MICHAEL PETERS

ABSTRACT. In this paper, we introduce and analyze a new low-rank multilevel strategy for the solution of random diffusion problems. Using a standard stochastic collocation scheme, we first approximate the infinite dimensional random problem by a deterministic parameter-dependent problem on a high-dimensional parameter domain. Given a hierarchy of finite element discretizations for the spatial approximation, we make use of a multilevel framework in which we consider the differences of the solution on two consecutive finite element levels in the collocation points. We then address the approximation of these high-dimensional differences by adaptive low-rank tensor techniques. This allows to equilibrate the error on all levels by exploiting analytic and algebraic properties of the solution at the same time. We arrive at an explicit representation in a low-rank tensor format of the approximate solution on the entire parameter domain, which can be used for, e.g., the direct and cheap computation of statistics. Numerical results are provided in order to illustrate the approach.

1. INTRODUCTION

In this article, we consider the random boundary value problem

$$(1) \qquad -\nabla \cdot \big(a(\omega)\nabla u(\omega)\big) = f \text{ in } D, \quad u(\omega) = 0 \text{ on } \partial D,$$

where $D \subset \mathbb{R}^d$ denotes a domain and $\omega \in \Omega$ is a random parameter, with Ω denoting the set of possible outcomes. As the solution u depends on the parameter ω, we aim at an efficient approximation of the solution map $\omega \mapsto u(\omega)$. The numerical solution of (1) has attracted quite some attention during the last decade, motivated by the need for quantifying the impact of uncertainties in PDE-based models.

The key idea of our novel approach is to combine a multilevel stochastic collocation framework with adaptive low-rank tensor techniques. This involves the following steps:

(1) A standard technique for random diffusion problems, the Karhunen-Loève expansion of the diffusion coefficient a is truncated after $N \in \mathbb{N}$ terms to turn (1) into a parametric PDE depending on N random parameters. This truncated problem is approximated by a stochastic collocation scheme.

(2) We use a hierarchy of finite element discretizations for discretizing the physical domain D and represent the solution u as a telescoping sum. The smoothness properties of the solution u are exploited to adapt the polynomial degrees in the stochastic collocation of the differences of u between two consecutive finite element levels. This allows us to choose a low polynomial degree for the fine spatial

discretization while using higher polynomial degrees only on coarser finite element levels.

(3) Because of the high dimensionality of the parameter domain, each difference in the multilevel sum needs to be evaluated in a large number of collocation points. We use adaptive low-rank tensor techniques to obtain good approximations from a relatively small number of samples. This allows us to exploit the algebraic structure of the solution with respect to the random parameters automatically while maintaining the accuracy of the multilevel scheme.

Both, multilevel and low-rank tensor approximation techniques, have been extensively studied for the solution of (1). In the following, we briefly describe some of the existing approaches.

A number of different multilevel techniques have been proposed that aim at equilibrating the errors of the spatial approximation and the approximation in the random parameter. If a statistics of the solution or a quantity of interest needs to be computed, multilevel quadrature methods, like the multilevel (quasi-)Monte Carlo method or even more general quadrature approaches, are feasible; we refer to [6, 16, 17, 20, 25, 27, 28] for instances of this approach. Closer to the setting considered in this paper, the work by Teckentrup et al. [44] proposes to directly interpolate the solution u itself in suitable collocation points in the parameter domain from a sparse index set. Given additional smoothness in the spatial variable, a spatial sparse-grid approximation can be incorporated, which leads to the multiindex stochastic collocation proposed in [24].

Low-rank tensor approximation techniques have turned out be a versatile tool for solving PDEs with random data; see [37, 38] and the references therein. In particular, a variety of low-rank approaches have been proposed to address the linear systems arising from a Galerkin discretization of (1); see, e.g., [10, 11, 12, 13, 14, 31, 32, 33, 36, 45]. Non-intrusive tensor-based approaches for uncertainty quantification can be built upon black box approximation techniques [2, 4, 39, 41].

To the best of our knowledge, there is little work on merging multilevel and tensor approximation techniques in uncertainty quantification. Recently, Lee and Elman [35] proposed a two-level scheme in the context of the Galerkin method for PDEs with random data. This scheme uses the solution from the coarse level to identify a dominant subspace in the domain of the random parameter, which in turn is used to speed up the solution on the fine level by avoiding costly low-rank truncations. The combination of multilevel and tensor approximation techniques proposed in this paper is conceptually different and is not restricted to this two level approach but allows for multiple levels.

The rest of this paper is organized as follows. In Section 2, we formulate the mathematical setting and recall the Karhunen-Loève expansion. Section 3 is concerned with the discretization of (1) in the physical and the stochastic domain. In Section 4, we describe an existing multilevel scheme and analyze the impact of perturbations on this scheme. Section 5 contains the main contribution of this paper, a novel combination of the multilevel scheme with a low-rank tensor approximation. Finally, Section 6 reports numerical results for PDEs with a random diffusion coefficient on the unit square featuring a variety of different stochastic diffusion coefficients.

Throughout this article, in order to avoid the repeated use of generic but unspecified constants, we indicate by $C \lesssim D$ that C can be bounded by a multiple of D, independently of parameters which C and D may depend on. Obviously, $C \gtrsim D$ is defined as $D \lesssim C$, and we write $C \approx D$ if there holds $C \lesssim D$ and $C \gtrsim D$.

2. Problem Setting

Let $D \subset \mathbb{R}^d$ denote a bounded Lipschitz domain. Typically, we have $d = 2, 3$. Moreover, let $(\Omega, \mathcal{F}, \mathbb{P})$ be a complete and separable probability space, where Ω is the set of outcomes, $\mathcal{F} \subset 2^\Omega$ is the σ-algebra of events, and $\mathbb{P} \colon \mathcal{F} \to [0, 1]$ is a probability measure on $\mathcal{F}$. We are interested in solving the following stochastic diffusion problem: Find $u \in L^2\big(\Omega; H_0^1(D)\big)$ such that

$$-\nabla \cdot \big(a(\omega)\nabla u(\omega)\big) = f, \quad \text{in } D,$$
$$u(\omega) = 0, \quad \text{on } \partial D,$$

holds $\mathbb{P}$-almost surely. Here and in the sequel, for a Banach space $\mathcal{X}$, we define the *Lebesgue-Bochner-space* $L^p(\Omega; \mathcal{X})$, $1 \le p \le \infty$ as the space of all equivalence classes of strongly measurable functions $v \colon \Omega \to \mathcal{X}$ whose norm

$$\|v\|_{L^p(\Omega;\mathcal{X})} := \begin{cases} \left(\int_\Omega \|v(\omega)\|_{\mathcal{X}}^p \mathrm{d}\mathbb{P}(\omega)\right)^{1/p}, & p < \infty \\ \operatorname*{ess\,sup}_{\omega \in \Omega} \|v(\omega)\|_{\mathcal{X}}, & p = \infty \end{cases}$$

is finite. If $p = 2$ and $\mathcal{X}$ is a separable Hilbert space, then the Bochner space is isomorphic to the tensor product space

$$L^2(\Omega; \mathcal{X}) \cong L^2(\Omega) \otimes \mathcal{X}.$$

Throughout this article, we shall assume that the load $f \in L^2(D)$ is purely deterministic. Still, by straightforward modifications it is also possible to deal with random load vectors, see, e.g., [1]. Additionally, for the sake of simplicity, we restrict ourselves here to the case of uniformly elliptic diffusion problems. This means that we assume the existence of constants $a_{\min} > 0$ and $a_{\max} < \infty$ that are independent of the parameter $\omega \in \Omega$ such that for almost every $x \in D$ there holds

(2) $$a_{\min} \le a(\omega, x) \le a_{\max} \quad \mathbb{P}\text{-almost surely.}$$

Nevertheless, we emphasize that the presented approach is directly transferable to diffusion problems, where the constants $a_{\min}$ and $a_{\max}$ might depend on $\omega \in \Omega$ and are only $\mathbb{P}$-integrable, as it is the case for log-normally distributed diffusion coefficients, cf. [30, 42]. Therefore, all results presented here remain valid in this case.

Typically, the diffusion coefficient is not directly feasible for numerical computations and has thus to be represented in a suitable way. To that end, one decomposes the diffusion coefficient with the aid of the *Karhunen-Loève expansion*.

Let the covariance kernel of $a(\omega, x)$ be defined by the positive semi-definite function

$$\mathcal{C}(x, x') := \int_\Omega \big(a(\omega, x) - \mathbb{E}[a](x)\big)\big(a(\omega, x') - \mathbb{E}[a](x')\big)\mathrm{d}\mathbb{P}.$$

Herein, the integral with respect to Ω has to be understood in terms of a Bochner integral, cf. [29]. One can show that $\mathcal{C}(x, x')$ is well defined if there holds $a \in L^2(\Omega; \mathcal{X})$. Now, let

$\{(\lambda_n, \varphi_n)\}_n$ denote the eigenpairs obtained by solving the eigenproblem for the diffusion coefficient's covariance, i.e.

$$\int_D \mathcal{C}(x, x')\varphi_n(x')\mathrm{d}x' = \lambda_n \varphi_n(x).$$

Then, the Karhunen-Loève expansion of $a(\omega, x)$ is given by

$$(3) \qquad a(\omega, x) = \mathbb{E}[a](x) + \sum_{n=1}^{\infty} \sqrt{\lambda_n} X_n(\omega)\varphi_n(x),$$

where $X_n \colon \Omega \to \Gamma_n \subset \mathbb{R}$ for $n = 1, 2, \dots$ are centered, pairwise uncorrelated and L^2-normalized random variables given by

$$X_n := \frac{1}{\sqrt{\lambda_n}} \int_D \big(a(\omega, x) - \mathbb{E}[a](x)\big)\varphi_n(x)\mathrm{d}x.$$

From condition (2), we directly infer, that the image of the random variables is a bounded set and that $\mathbb{E}[a](\mathbf{x}) > 0$. Thus, without loss of generality, we assume that $\Gamma_n = [-1, 1]$. The important cases, which we wish to study here, are the uniformly distributed case, i.e. $X_n \sim \mathcal{U}([-1, 1])$ and the log-uniformly distributed case which means that we have diffusion coefficient of the form $\exp\big(a(\omega, x)\big)$, where $a(\omega, x)$ is given as in the uniformly distributed case and satisfies (2).

Although, we have separated by now the spatial and the stochastic influences in the diffusion coefficient, we are still facing an infinite sum. Nevertheless, for numerical issues, this sum may be truncated appropriately. The impact of truncating the Karhunen-Loève expansion on the solution is bounded by

$$\|u - u_N\|_{L^2(\Omega; H_0^1(D))} \lesssim \|a - a_N\|_{L^2(\Omega; L^\infty(D))} = \varepsilon(N),$$

where $\varepsilon(N) \to 0$ montonically as $N \to \infty$, see e.g. [8, 43]. Herein, we set

$$a_N(\omega, x) := \mathbb{E}[a](x) + \sum_{n=1}^{N} \sqrt{\lambda_n} X_n(\omega)\varphi_n(x),$$

and u_N is the solution to

$$-\nabla \cdot \big(a_N(\omega)\nabla u_N(\omega)\big) = f \text{ in } D, \quad u(\omega) = 0 \text{ on } \partial D.$$

Note that these estimates relate to the log-normal and the uniformly distributed cases. But they also directly transfer to the log-uniform case.

Assuming additionally, that the $\{X_n\}_n$ are independent and exhibit densities $\rho_n \colon \Gamma_N \to \mathbb{R}_+$ with respect to the Lebesgue measure, we end up, with the parametric diffusion problem: Find $u_N \in L_\rho^2\big(\Gamma; H_0^1(D)\big)$

$$(4) \qquad\qquad -\nabla \cdot \big(a_N(\boldsymbol{y})\nabla u_N(\boldsymbol{y})\big) = f \text{ in } D,$$

where $\rho := \rho_1(y_1)\cdots\rho_N(y_N)$, $\Gamma := \times_{n=1}^{N} \Gamma_n$ and $\boldsymbol{y} = \boldsymbol{y}(\omega) := [y_1(\omega), \dots, y_N(\omega)] \in \Gamma$. Herein, the space $L_\rho^2\big(\Gamma; H_0^1(D)\big)$ is endowed with the norm

$$\|v\|_{L_\rho^2(\Gamma; H_0^1(D))} := \left(\int_\Gamma \|v(\boldsymbol{y})\|_{H_0^1(D)}^2 \rho(\boldsymbol{y})\mathrm{d}\boldsymbol{y}\right)^{1/2}.$$

Note that we have $\rho_n = 1/2$ for the case of $X_n \sim \mathcal{U}([-1,1])$. In view of the polynomial interpolation with respect to the parameter $\boldsymbol{y} \in \Gamma$, we shall finally introduce for a Banach space $\mathcal{X}$ the space

$$C^0(\Gamma; \mathcal{X}) := \left\{ v \colon \Gamma \to \mathcal{X} : v \text{ is continuous and } \sup_{\boldsymbol{y} \in \Gamma} \|v(\boldsymbol{y})\|_{\mathcal{X}} < \infty \right\}.$$

3. DISCRETIZATION

Later on, a standard *stochastic collocation* scheme, cf. [1], is used for the stochastic discretization of the differences of the solutions to the parametric diffusion problem (4) on consecutive grids. To that end, we use tensor product polynomial interpolation in the parameter space Γ and a finite element approximation in the physical domain D.

3.1. **Polynomial Interpolation.** Let $\mathcal{P}_{\boldsymbol{p}}(\Gamma) \subset L_\rho^2(\Gamma)$ denote the span of tensor product polynomials with degree at most $\boldsymbol{p} = (p_1, \ldots, p_N)$, i.e.,

$$\mathcal{P}_{\boldsymbol{p}}(\Gamma) = \bigotimes_{n=1}^{N} \mathcal{P}_{p_n}(\Gamma_n)$$

with

$$\mathcal{P}_{p_n}(\Gamma_n) = \text{span}\{y_n^m : m = 0, \ldots, p_n\}, \quad n = 1, \ldots, N.$$

Given interpolation points $y_{n,k_n} \in \Gamma_n$, $k_n = 0, \ldots, p_n$, the Lagrange basis for $\mathcal{P}_{p_n}(\Gamma_n)$ is defined by $\{l_{n,k_n} \in \mathcal{P}_{p_n}(\Gamma_n) : l_{n,k_n}(y_{n,j_n}) = \delta_{k_n,j_n}, k_n = 0, \ldots, p_n\}$. By a tensor product construction, we obtain the Lagrange basis $\{l_{\boldsymbol{k}}\}$ for $\mathcal{P}_{\boldsymbol{p}}(\Gamma)$ where

$$l_{\boldsymbol{k}}(\boldsymbol{y}) := \prod_{n=1}^{N} l_{n,k_n}(y_n)$$

for a multiindex $\boldsymbol{k} = (k_1, \ldots, k_N) \in \mathcal{K}_{\boldsymbol{p}}$ with

$$\mathcal{K}_{\boldsymbol{p}} := \{(k_1, \ldots, k_N) \in \mathbb{N}_0^N : k_n = 0, \ldots, p_n, \, n = 1, \ldots, N\}.$$

For all functions $v \in C^0\big(\Gamma; H_0^1(D)\big)$, the tensor product interpolation points $\boldsymbol{y}_{\boldsymbol{k}} := (y_{1,k_1}, \ldots, y_{N,k_N}) \in \Gamma$ give rise to the interpolation operator

$$\mathcal{I}_{\boldsymbol{p}} \colon C^0\big(\Gamma; H_0^1(D)\big) \to \mathcal{P}_{\boldsymbol{p}}(\Gamma) \otimes H_0^1(D)$$

defined by

$$(5) \qquad \mathcal{I}_{\boldsymbol{p}}[v](\boldsymbol{y}) = \sum_{\boldsymbol{k} \in \mathcal{K}_{\boldsymbol{p}}} v(\boldsymbol{y}_{\boldsymbol{k}}) l_{\boldsymbol{k}}(\boldsymbol{y}).$$

With regard to (4), our goal is to approximate the solution u_N by

$$u_N(\boldsymbol{y}) \approx \mathcal{I}_{\boldsymbol{p}}[u_N](\boldsymbol{y}) = \sum_{\boldsymbol{k} \in \mathcal{K}_{\boldsymbol{p}}} u_N(\boldsymbol{y}_{\boldsymbol{k}}) l_{\boldsymbol{k}}(\boldsymbol{y}).$$

In order to obtain the coefficients $u_N(\boldsymbol{y}_{\boldsymbol{k}})$, we have to solve

$$(6) \qquad -\nabla \cdot \big(a_N(\boldsymbol{y}_{\boldsymbol{k}}) \nabla u_N(\boldsymbol{y}_{\boldsymbol{k}})\big) = f \text{ in } D, \quad u_N(\boldsymbol{y}_{\boldsymbol{k}}) = 0 \text{ on } \partial D,$$

for all $\boldsymbol{y}_{\boldsymbol{k}}$ with $\boldsymbol{k} \in \mathcal{K}_{\boldsymbol{p}}$. For each $\boldsymbol{k} \in \mathcal{K}_{\boldsymbol{p}}$, (6) is a deterministic diffusion problem on D which can be approximated by the finite element method.

3.2. **Interpolation Error.** To study the impact of the interpolation error, we have to take the smoothness of u_N with respect to the parameter $\boldsymbol{y} \in \Gamma$ into account. It is well known, see, e.g., [9, 30], that u_N satisfies the decay estimate

$$(7) \qquad \left\| \partial_{\boldsymbol{y}}^{\boldsymbol{\alpha}} u_N(\boldsymbol{y}) \right\|_{H_0^1(D)} \leq C |\boldsymbol{\alpha}|! c^{|\boldsymbol{\alpha}|} \boldsymbol{\gamma}^{\boldsymbol{\alpha}} \|f\|_{L^2(D)}, \text{ where } \gamma_n := \sqrt{\lambda_n} \|\varphi\|_{L^\infty(D)},$$

cf. (3), for some constants $C, c > 0$. In the sequel, we consider the interpolation based on the Chebyshev nodes

$$\eta_k := \cos\left(\frac{2k+1}{2(p+1)} \pi \right) \in [-1, 1], \quad k = 0, \ldots, p.$$

The related uni-directional interpolation operator shall be denoted by

$$\mathcal{I}_p : C([-1,1]) \to \mathcal{P}_p, \quad v(x) \mapsto \sum_{k=0}^p v(\eta_k) l_k(x).$$

It satisfies for a function $v \in C^{p+1}([-1,1])$ the well known interpolation error estimate

$$\left| v(x) - \sum_{k=0}^p v(\eta_k) l_k(x) \right| \leq \frac{1}{2^p (p+1)!} \max_{\xi \in [-1,1]} \left| v^{(p+1)}(\xi) \right|$$

and the stability estimate

$$\left\| \sum_{k=0}^p v(\eta_k) l_k(x) \right\|_{C^0([-1,1])} \leq \left(\frac{2}{\pi} \log(p+1) + 1 \right) \|v\|_{C^0([-1,1])},$$

see, e.g., [40]. Therefore, we obtain by tensor product construction the stability estimate for $\mathcal{I}_{\boldsymbol{p}}$ according to

$$\left\| \mathcal{I}_{\boldsymbol{p}}[v] \right\|_{C^0(\Gamma; H_0^1(D))} \leq C_s(\boldsymbol{p}) \|v\|_{C^0(\Gamma; H_0^1(D))}$$

with

$$C_s(\boldsymbol{p}) := \prod_{i=1}^N \left(\frac{2}{\pi} \log(p_i + 1) + 1 \right).$$

Obviously, the stability constant will grow exponentially as $N \to \infty$. Nevertheless, this case is not considered here. Moreover, we emphasize that there exist regimes, where the stability constant is bounded. If the error is, for example, measured in $L_\rho^2(\Gamma; H_0^1(D))$ and the interpolation points are chosen as the roots of the orthogonal polynomials with respect to the densities ρ_n, then the corresponding stability estimate holds with $C_s(\boldsymbol{p}) = 1$, cf. [1]. Still, without the orthogonality property, there exist also bounds of the stability constant for Chebyshev points, if the error is measured in $L^1(\Gamma; H_0^1(D))$, see [15]. Nevertheless, in order to obtain a black box interpolation, which is independent of the particular density function, we will employ here the Chebyshev points and measure the error with respect to $C^0(\Gamma; H_0^1(D))$ at the cost of a stability constant that is not robust with respect to the polynomial degree.

Thus, we obtain the following interpolation result for the solution u_N to (4), which is a straightforward modification of the related result in [22].

Theorem 1. *Let $c\gamma_k < 2$. Then, given that*

$$p_k = \left\lceil \frac{\log(\varepsilon)}{\log(c\gamma_k/2)} \right\rceil - 1,$$

the polynomial interpolation satisfies the error estimate

$$\big\|u_N(\boldsymbol{y}) - \mathcal{I}_{\boldsymbol{p}}[u_N](\boldsymbol{y})\big\|_{H_0^1(D)} \lesssim \varepsilon C(\boldsymbol{p})\|f\|_{L^2(D)}$$

for some constant $C(\boldsymbol{p})$.

Proof. There holds by (7) and the repeated application of the triangle inequality that

$$\big\|u_N(\boldsymbol{y}) - \mathcal{I}_{\boldsymbol{p}}[u_N](\boldsymbol{y})\big\|_{H_0^1(D)}$$

$$\leq \sum_{k=1}^{N} \big\|\big(\mathcal{I}_{p_1} \otimes \ldots \otimes \mathcal{I}_{p_{k-1}} \otimes (\mathrm{Id} - \mathcal{I}_{p_k}) \otimes \mathrm{Id} \otimes \ldots \otimes \mathrm{Id}\,\big)u_N(\boldsymbol{y})\big\|_{H_0^1(D)}$$

$$\leq \sum_{k=1}^{N} \bigg[\prod_{m=1}^{k-1}\Big(\frac{2}{\pi}\log(p_m+1)+1\Big)\bigg]\bigg[\frac{1}{2^{p_k}(p_k+1)!}C(p_k+1)!c^{p_k+1}\gamma_k^{p_k+1}\bigg]\|f\|_{L^2(D)}$$

$$= \sum_{k=1}^{N} \bigg[\prod_{m=1}^{k-1}\Big(\frac{2}{\pi}\log(p_m+1)+1\Big)\bigg]\bigg[2\Big(\frac{c\gamma_k}{2}\Big)^{p_k+1}Cc\bigg]\|f\|_{L^2(D)}.$$

Thus, with

$$p_k = \left\lceil \frac{\log(\varepsilon)}{\log(c\gamma_k/2)} \right\rceil - 1,$$

we obtain

$$\big\|u_N(\boldsymbol{y}) - \mathcal{I}_{\boldsymbol{p}}[u_N](\boldsymbol{y})\big\|_{H_0^1(D)} \leq Cc\varepsilon\bigg(\sum_{k=1}^{N}\bigg[\prod_{m=1}^{k-1}\Big(\frac{2}{\pi}\log(p_m+1)+1\Big)\bigg]\bigg)\|f\|_{L^2(D)}.$$

$\square$

Remark 2. The constant $C(\boldsymbol{p})$ from the previous theorem can be bounded according to

$$C_s(\boldsymbol{p}) \leq C(\boldsymbol{p}) \leq (N+1)C_s(\boldsymbol{p}).$$

where we recall that $C_s(\boldsymbol{p})$ denotes the stability constant of $\mathcal{I}_{\boldsymbol{p}}$. Thus, $C(\boldsymbol{p})$ also potentially grows exponentially as $N \to \infty$.

3.3. **Finite Element Approximation.** In order to compute the coefficients $u_N(\boldsymbol{y_k})$ in (6), we consider an approximation by the finite element method. To this end, let $\mathcal{T}_0 = \{\tau_{0,k}\}$ be a coarse grid triangulation of the domain D. Then, for $\ell \geq 1$, a uniform and shape regular triangulation $\mathcal{T}_\ell = \{\tau_{\ell,k}\}$ is recursively obtained by uniformly refining each element $\tau_{\ell-1,k}$ into 2^d elements with diameter $h_\ell \approx 2^{-\ell}$. We define the space of piecewise linear finite elements according to

(8) $$\mathcal{S}_\ell^1(D) := \{v \in C(D) : v|_{\partial D} = 0 \text{ and } v|_\tau \in \Pi_1 \text{ for all } \tau \in \mathcal{T}_\ell\} \subset H_0^1(D),$$

where Π_1 denotes the space of all polynomials of total degree 1. Then, the finite element approximations $u_{N,\ell}(\boldsymbol{y_k}) \in \mathcal{S}_\ell^1(D)$ to the coefficients $u_N(\boldsymbol{y_k})$ satisfy the following well known error estimate.

Lemma 3. *Let the domain D be convex and $f \in L^2(D)$. Then, for $\boldsymbol{y} \in \Gamma$, the finite element solution $u_{N,\ell}(\boldsymbol{y}) \in \mathcal{S}_\ell^1(D)$ of the diffusion problem (4) satisfies the error estimate*

(9) $$\|u_N(\boldsymbol{y}) - u_{N,\ell}(\boldsymbol{y})\|_{H^1(D)} \lesssim 2^{-\ell}\|u_N(\boldsymbol{y})\|_{H^2(D)} \lesssim 2^{-\ell}\|f\|_{L^2(D)}.$$

Note that we restrict ourselves here to the situation of piecewise linear finite elements. Nevertheless, by applying obvious modifications, the presented results remain valid also for higher order finite elements. Moreover, for the sake of simplicity, we consider here nested sequences of finite element spaces, i.e.,

$$(10) \qquad \mathcal{S}_0^1(D) \subset \mathcal{S}_1^1(D) \subset \dots.$$

This is not a requirement, as has been discussed in [20].

3.4. **Stochastic Collocation Error.** By a tensor product argument, the combination of the finite element approximation in the spatial variable and the interpolation in the parameter yields the following approximation result.

Theorem 4. *Let the polynomial degree $\boldsymbol{p}$ be chosen such that there holds*

$$\big\| u_N(\boldsymbol{y}) - \mathcal{I}_{\boldsymbol{p}}[u_N](\boldsymbol{y}) \big\|_{H_0^1(D)} \lesssim 2^{-\ell} C(\boldsymbol{p}) \|f\|_{L^2(D)},$$

where $u_{N,\ell}(\boldsymbol{y})$ is the finite element approximation to $u_N(\boldsymbol{y})$ on level ℓ that fulfills (9). Then, the fully discrete approximation $\mathcal{I}_{\boldsymbol{p}}[u_{N,\ell}] \in \mathcal{P}_{\boldsymbol{p}}(\Gamma) \otimes \mathcal{S}_j^1(D)$ satisfies the error estimate

$$\big\| u_N(\boldsymbol{y}) - \mathcal{I}_{\boldsymbol{p}}[u_{N,\ell}](\boldsymbol{y}) \big\|_{H_0^1(D)} \lesssim 2^{-\ell} \big(C(\boldsymbol{p}) + C_s(\boldsymbol{p}) \big) \|f\|_{L^2(D)},$$

where $C_s(\boldsymbol{p})$ denotes the stability constant of $\mathcal{I}_{\boldsymbol{p}}$.

4. Multilevel Approximation

In the previous section, we have introduced the classical stochastic collocation as it has been proposed in, e.g., [1]. The related error estimate is in this case based on a tensor product argument between the spatial approximation and the discretization of the parameter. Now, the idea of the related multilevel approximation is to perform an error equilibration as it is known from *sparse* tensor product approximations, cf. [7].

4.1. **Multilevel Scheme.** We start by representing the finite element approximation $u_{N,L}(\boldsymbol{y})$ for a maximal level $L \in \mathbb{N}$ by the telescoping sum

$$u_{N,L}(\boldsymbol{y}) = \sum_{\ell=0}^{L} \big(u_{N,\ell}(\boldsymbol{y}) - u_{N,\ell-1}(\boldsymbol{y}) \big) \quad \text{with } u_{N,-1} := 0.$$

Instead of applying the interpolation operator in the parameter $\boldsymbol{y} \in \Gamma$ with a fixed degree $\boldsymbol{p}$, we adapt the degree to the finite element approximation level and obtain the multilevel approximation

$$(11) \qquad u_N(\boldsymbol{y}) \approx u_{N,L}^{\mathrm{ML}}(\boldsymbol{y}) := \sum_{\ell=0}^{L} \mathcal{I}_{\boldsymbol{p}^{(\ell)}} \big[u_{N,\ell} - u_{N,\ell-1} \big](\boldsymbol{y}).$$

The goal is now to choose the polynomial degrees $\{\boldsymbol{p}^{(\ell)}\}$ antipodal to the refinement level ℓ of the finite element approximation and to equilibrate a high spatial accuracy with a relatively low polynomial degree. In order to facilitate this, it is crucial to have the following *mixed regularity* estimate for u_N. There holds

$$\big\| \partial_{\boldsymbol{y}}^{\boldsymbol{\alpha}} u(\boldsymbol{y}) \big\|_{H^2(D)} \leq C |\boldsymbol{\alpha}|! c^{|\boldsymbol{\alpha}|} \tilde{\boldsymbol{\gamma}}^{\boldsymbol{\alpha}} \|f\|_{L^2(D)}, \text{ where } \tilde{\gamma}_k := \sqrt{\lambda_k} \|\varphi_k\|_{W^{1,\infty}(D)},$$

cp. (3), for some constants $C, c > 0$. See [9] for a proof of this statement in the affine case and [30] for the log-normal case. The estimate for the log-uniform case can be derived with the same techniques that are applied in these works. From this estimate, one can derive the parametric smoothness of the Galerkin error. This is stated by the following lemma which is, e.g., proven in [26, 34].

Lemma 5. *For the error of the Galerkin projection, there holds the estimate*

$$\left\|\partial_{\boldsymbol{y}}^{\boldsymbol{\alpha}}(u_N - u_{N,\ell})(\boldsymbol{y})\right\|_{H^1(D)} \lesssim 2^{-\ell} |\boldsymbol{\alpha}|! c^{|\boldsymbol{\alpha}|} \boldsymbol{\gamma}^{\boldsymbol{\alpha}} \|f\|_{L^2(D)} \quad \text{for all } |\boldsymbol{\alpha}| \geq 0$$

with a constant $c > 0$ depending on $a_{\min}$ and $a_{\max}$, where $\gamma_k := \|\sqrt{\lambda_k}\varphi_k\|_{W^{1,\infty}(D)}$ from (3) and $\boldsymbol{\gamma} := (\gamma_1, \ldots, \gamma_m)$.

With this lemma at hand, it is easy to derive the following error estimate in complete analogy to the proof of Theorem 1.

Theorem 6. *Let the degree $\boldsymbol{p}^{(\ell')} \in \mathbb{N}^N$ be such that the interpolation error is $C(\boldsymbol{p}^{(\ell')})\varepsilon \approx 2^{-\ell'}$. Then, there holds the error estimate*

$$(12) \qquad \left\|(\mathrm{Id} - \mathcal{I}_{\boldsymbol{p}^{(\ell')}})[u_N - u_{N,\ell}](\boldsymbol{y})\right\|_{H^1(D)} \lesssim 2^{-(\ell+\ell')}\|f\|_{L^2(D)}.$$

Theorem 7. *Let $\{\boldsymbol{p}^{(\ell')}\} \in \mathbb{N}^N$ be a sequence of polynomial degrees, that give rise to an error estimate of the form (12) with $\ell' = L - \ell$, where $u_N \in L_\rho^2(\Gamma, H_0^1(D))$ is the solution to (4) that satisfies (9). Then, the error of the multilevel approximation (11) is bounded by*

$$(13) \qquad \left\|u_N(\boldsymbol{y}) - \sum_{\ell=0}^{L} \mathcal{I}_{\boldsymbol{p}^{(\ell)}}[u_{N,\ell} - u_{N,\ell-1}](\boldsymbol{y})\right\|_{H_0^1(D)} \lesssim 2^{-L} L \|f\|_{L^2(D)}.$$

Proof. We shall apply the following multilevel splitting of the error
$$(14)$$
$$\left\|u_N(\boldsymbol{y}) - \sum_{\ell=0}^{L} \mathcal{I}_{\boldsymbol{p}^{(\ell)}}[u_{N,\ell} - u_{N,\ell-1}](\boldsymbol{y})\right\|_{H_0^1(D)}$$
$$= \left\|u_N(\boldsymbol{y}) - u_{N,L}(\boldsymbol{y}) + \sum_{\ell=0}^{L}(u_{N,\ell} - u_{N,\ell-1})(\boldsymbol{y}) - \sum_{\ell=0}^{L} \mathcal{I}_{\boldsymbol{p}^{(\ell)}}[u_{N,\ell} - u_{N,\ell-1}](\boldsymbol{y})\right\|_{H_0^1(D)}$$
$$\leq \left\|u_N(\boldsymbol{y}) - u_{N,L}(\boldsymbol{y})\right\|_{H_0^1(D)} + \sum_{\ell=0}^{L} \left\|(\mathrm{Id} - \mathcal{I}_{\boldsymbol{p}^{(\ell')}})[u_{N,\ell} - u_{N,\ell-1}](\boldsymbol{y})\right\|_{H_0^1(D)}.$$

The first term just reflects the finite element approximation error and is thus bounded by (9). Thanks to (12), the term inside the sum satisfies

$$\left\|(\mathrm{Id} - \mathcal{I}_{\boldsymbol{p}^{(\ell')}})[u_{N,\ell} - u_{N,\ell-1}](\boldsymbol{y})\right\|_{H_0^1(D)}$$
$$\leq \left\|(\mathrm{Id} - \mathcal{I}_{\boldsymbol{p}^{(\ell')}})[u_N - u_{N,\ell}](\boldsymbol{y})\right\|_{H_0^1(D)} + \left\|(\mathrm{Id} - \mathcal{I}_{\boldsymbol{p}^{(\ell')}})[u_N - u_{N,\ell-1}](\boldsymbol{y})\right\|_{H_0^1(D)}$$
$$\lesssim 2^{-(\ell+L-\ell)}\|f\|_{L^2(D)} + 2^{-(\ell-1+L-\ell)}\|f\|_{L^2(D)} \lesssim 2^{-L}\|f\|_{L^2(D)}.$$

Thus, we can estimate (14) as

$$\left\|u_N(\boldsymbol{y}) - \sum_{\ell=0}^{L} \mathcal{I}_{\boldsymbol{p}^{(\ell)}}\left[u_{N,\ell} - u_{N,\ell-1}\right](\boldsymbol{y})\right\|_{H_0^1(D)} \lesssim 2^{-L}\|f\|_{L^2(D)} + \sum_{\ell=0}^{L} 2^{-L}\|f\|_{L^2(D)}$$

$$\leq 2^{-L}(L+2)\|f\|_{L^2(D)}.$$

This completes the proof. □

4.2. **Perturbed Multilevel Scheme.** The multilevel scheme from above relies on the exact evaluation of the differences

$$\delta_{N,\ell} := u_{N,\ell} - u_{N,\ell-1}$$

in the interpolation points $\boldsymbol{y}_{\boldsymbol{k}}^{(\ell)} \in \mathcal{K}_{\boldsymbol{p}^{(\ell)}} \subset \Gamma$ on each level ℓ. We now slightly relax this assumption and consider perturbations

$$(15) \qquad\qquad \tilde{\delta}_{N,\ell,\boldsymbol{k}} \approx \delta_{N,\ell}(\boldsymbol{y}_{\boldsymbol{k}}^{(\ell)}), \qquad \boldsymbol{k} \in \mathcal{K}_{\boldsymbol{p}^{(\ell)}},$$

and the associated perturbed interpolation

$$\tilde{\delta}_{N,\ell}(\boldsymbol{y}) := \sum_{\boldsymbol{k}\in\mathcal{K}_{\boldsymbol{p}^{(\ell)}}} \tilde{\delta}_{N,\ell,\boldsymbol{k}} l_{\boldsymbol{k}}(\boldsymbol{y}).$$

In view of (11), the perturbed multilevel approximation then reads

$$(16) \qquad\qquad \tilde{u}_{N,L}^{\mathbb{ML}}(\boldsymbol{y}) := \sum_{\ell=0}^{L} \tilde{\delta}_{N,\ell}(\boldsymbol{y}).$$

For each level ℓ, we have the stability estimate

$$\|\tilde{\delta}_{N,\ell}(\boldsymbol{y}) - \mathcal{I}_{\boldsymbol{p}^{(\ell)}}[\delta_{N,\ell}](\boldsymbol{y})\|_{H^1(D)} \leq C_s(\boldsymbol{p}^{(\ell)}) \max_{\boldsymbol{k}\in\mathcal{K}_{\boldsymbol{p}^{(\ell)}}} \|\tilde{\delta}_{N,\ell,\boldsymbol{k}} - \delta_{N,\ell}(\boldsymbol{y}_{\boldsymbol{k}}^{(\ell)})\|_{H^1(D)}.$$

From Theorem 7, we immediately derive the following lemma.

Lemma 8. *Assume that for all $\boldsymbol{k} \in \mathcal{K}_{\boldsymbol{p}^{(\ell)}}$ the perturbations from (15) fulfill*

$$\left\|\tilde{\delta}_{N,\ell,\boldsymbol{k}} - \delta_{N,\ell}(\boldsymbol{y}_{\boldsymbol{k}}^{(\ell)})\right\|_{H^1(D)} \lesssim 2^{-L}\|f\|_{L^2(D)}.$$

Then

$$\left\|u_N(\boldsymbol{y}) - \tilde{u}_{N,L}^{\mathbb{ML}}(\boldsymbol{y})\right\|_{H^1(D)} \lesssim 2^{-L} L \|f\|_{L^2(D)}.$$

A particular perturbation based on low-rank truncations will be considered in the following section.

5. Low-Rank Tensor Approximation

The main computational challenge in the multilevel scheme presented above is the evaluation of the differences $\delta_{N,\ell}(\boldsymbol{y}_{\boldsymbol{k}}^{(\ell)})$ for all $\boldsymbol{k} \in \mathcal{K}_{\boldsymbol{p}^{(\ell)}}$. To address high parameter dimensions N, we suggest to approximate these differences in a low-rank tensor format.

Let $n_\ell := \dim \mathcal{S}_\ell^1(D)$ for the finite element space from (8) and let $\{\psi_{\ell,i} \in \mathcal{S}_\ell^1(D) : i = 1,\ldots,n_\ell\}$ be an orthonormal basis of $\mathcal{S}_\ell^1(D)$ with respect to the H_0^1 inner product, i.e.,

$$\langle\psi_{\ell,i}, \psi_{\ell,j}\rangle_{H_0^1(D)} = 0, \qquad i \neq j,$$

and $\|\psi_{\ell,i}\|_{H_0^1(D)} = 1$. Given the nestedness assumption (10), we have $\delta_{N,\ell}(\boldsymbol{y}) \in \mathcal{S}_\ell^1(D)$. We can hence write

$$
(17) \qquad \delta_{N,\ell}(\boldsymbol{y}) = \sum_{i=1}^{n_\ell} \mathbf{u}_i^{(\ell)}(\boldsymbol{y})\psi_{\ell,i}
$$

with $\mathbf{u}^{(\ell)}(\boldsymbol{y}) \in \mathbb{R}^{n_\ell}$. Let now $K_\ell := \#\mathcal{K}_{\boldsymbol{p}^{(\ell)}}$ and define $\mathbf{X}^{(\ell)} \in \mathbb{R}^{K_\ell \cdot n_\ell}$ as

$$
(18) \qquad \mathbf{X}_{(\boldsymbol{k},i)}^{(\ell)} := \mathbf{u}_i^{(\ell)}(\boldsymbol{y}_{\boldsymbol{k}}^{(\ell)}), \qquad \boldsymbol{k} \in \mathcal{K}_{\boldsymbol{p}^{(\ell)}}.
$$

In the following, we interpret the vector $\mathbf{X}^{(\ell)}$ as a tensor of order $N+1$ and size

$$
(p_1^{(\ell)} + 1) \times \cdots \times (p_N^{(\ell)} + 1) \times n_\ell
$$

and use low-rank tensor methods to construct a data-sparse approximation $\tilde{\mathbf{X}}^{(\ell)} \approx \mathbf{X}^{(\ell)}$. In particular, we make use of the hierarchical tensor format introduced in [23] and analyzed in [18].

5.1. **Hierarchical Tensor Format.** In the following, we consider tensors $\mathbf{X} \in \mathbb{R}^{\mathcal{J}}$ of order $d \in \mathbb{N}$ over general product index sets $\mathcal{J} = \mathcal{J}_1 \times \ldots \times \mathcal{J}_d$. We first need the concept of the matricization of a tensor.

Definition 9. Let $\mathcal{D} := \{1, \ldots, d\}$. Given a subset $t \subset \mathcal{D}$ with complement $[t] := \mathcal{D} \setminus t$, the *matricization*

$$
\mathcal{M}_t : \mathbb{R}^{\mathcal{J}} \to \mathbb{R}^{\mathcal{J}_t} \otimes \mathbb{R}^{\mathcal{J}_{[t]}}, \qquad \mathcal{J}_t := \bigtimes_{i \in t} \mathcal{J}_i, \quad \mathcal{J}_{[t]} := \bigtimes_{i \in [t]} \mathcal{J}_i,
$$

of a tensor $\mathbf{X} \in \mathbb{R}^{\mathcal{J}}$ is defined by its entries

$$
\mathcal{M}_t(\mathbf{X})_{(j_i)_{i \in t},(j_i)_{i \in [t]}} := \mathbf{X}_{(j_1,\ldots,j_d)}, \qquad (j_1, \ldots, j_d) \in \mathcal{J}.
$$

The subsets $t \subset \mathcal{D}$ are organized in a binary *dimension tree* $T_{\mathcal{D}}$ with root $\mathcal{D}$ such that each node $t \in T_{\mathcal{D}}$ is non-empty and each $t \in T_{\mathcal{D}}$ with $\#t \geq 2$ is the disjoint union of its sons $t_1, t_2 \in T_{\mathcal{D}}$, cf. Figure 1.
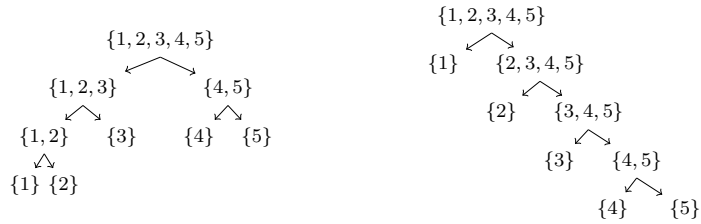

FIGURE 1. Dimension trees $T_{\mathcal{D}}$ for $d = 5$. Left: Balanced tree. Right: Linear tree.

Definition 10. Let $T_{\mathcal{D}}$ be a dimension tree. The *hierarchical rank* $\boldsymbol{r} := (r_t)_{t \in T_{\mathcal{D}}}$ of a tensor $\mathbf{X} \in \mathbb{R}^{\mathcal{J}}$ is defined by

$$
r_t := \operatorname{rank}\big(\mathcal{M}_t(\mathbf{X})\big), \quad t \in T_{\mathcal{D}}.
$$

For a given hierarchical rank $\boldsymbol{r} := (r_t)_{t \in T_{\mathcal{D}}}$, the *hierarchical format* $\mathcal{H}_{\boldsymbol{r}}$ is defined by

$$
\mathcal{H}_{\boldsymbol{r}} := \big\{\mathbf{X} \in \mathbb{R}^{\mathcal{J}} : \operatorname{rank}\big(\mathcal{M}_t(\mathbf{X})\big) \leq r_t, \, t \in T_{\mathcal{D}}\big\}.
$$

Given a tensor $\mathbf{X} \in \mathcal{H}_r$, Definition 10 implies that for all $t \in T_\mathcal{D}$ we can choose (orthogonal) matrices $\mathbf{U}_t \in \mathbb{R}^{\mathcal{J}_t \times r_t}$ such that $\mathrm{range}(\mathbf{U}_t) = \mathrm{range}\left(\mathcal{M}_t(\mathbf{X})\right)$. Moreover, for every non-leaf node $t \in T_\mathcal{D}$ with sons $t_1, t_2 \in T_\mathcal{D}$, there exists a *transfer tensor* $\mathbf{B}_t \in \mathbb{R}^{r_t \times r_{t_1} \times r_{t_2}}$ such that

$$(19) \qquad (\mathbf{U}_t)_{\cdot,s} = \sum_{s_1=1}^{r_{t_1}} \sum_{s_2=1}^{r_{t_2}} (\mathbf{B}_t)_{(s,s_1,s_2)} (\mathbf{U}_{t_1})_{\cdot,s_1} \otimes (\mathbf{U}_{t_2})_{\cdot,s_2}, \qquad s = 1, \ldots, r_t,$$

where $(\mathbf{U}_t)_{\cdot,s}$ denotes the sth column of $\mathbf{U}_t$. At the root node $t = \mathcal{D}$, we identify the tensor $\mathbf{X}$ with the column matrix $\mathbf{U}_\mathcal{D} \in \mathbb{R}^{\mathcal{J} \times 1}$.

The recursive relation (19) is key to represent the tensor $\mathbf{X}$ compactly. For all leaf nodes $t \in T_\mathcal{D}$, we explicitly store the matrices $\mathbf{U}_t$, whereas for all inner nodes $t \in T_\mathcal{D}$ only the transfers tensors $\mathbf{B}_t$ are required. The complexity for the hierarchical representation sums up to $\mathcal{O}(dr^3 + drn)$, where $r := r_{\max} = \max_{t \in T_\mathcal{D}} r_t$, $n := \max_{i \in \mathcal{D}} \#\mathcal{J}_i$. The *effective rank* r_{eff} is the real positive number such that $(d-1)r_{\mathrm{eff}}^3 + dr_{\mathrm{eff}}n$ is the actual storage cost for a tensor in $\mathcal{H}_r$.

In the multilevel scheme introduced above, the tensor $\mathbf{X}^{(\ell)}$ from (18) is defined via the numerical solution of the original PDE on levels ℓ and $\ell - 1$ at all collocation points. This means that an explicit computation of $\mathbf{X}^{(\ell)}$ in terms of all its entries would only be possible for small length N of the Karhunen-Loève expansion and moderate polynomial degrees $\boldsymbol{p}^{(\ell)}$. To overcome this limitation, we suggest to approximate $\mathbf{X}^{(\ell)}$ directly in the hierarchical tensor format $\mathcal{H}_r$ by the so-called *cross approximation* technique introduced in [3].

5.2. **Cross Approximation Technique.** The main idea of tensor cross approximation is to exploit the inherent low-rank structure directly by the evaluation of a (small) number of well-chosen tensor entries. Prior numerical experiments indicate that cross approximation works particularly well for tensors of small size in each direction $i = 1, \ldots, d$. Considering the tensor $\mathbf{X} = \mathbf{X}^{(\ell)}$ from (18), we observe that the size n_ℓ in direction $d = N + 1$ becomes rather large for higher levels ℓ such that the cross approximation technique cannot be applied directly. We therefore use the following variant consisting of three steps:

Step 1.: Find an (orthogonal) matrix $\mathbf{V} \in \mathbb{R}^{n_\ell \times r_d}$ such that

$$\mathcal{M}_{\{d\}}(\mathbf{X}) \approx \mathbf{V}\mathbf{V}^\top \mathcal{M}_{\{d\}}(\mathbf{X}).$$

Step 2.: Define a tensor $\mathbf{Y} \in \mathbb{R}^{\mathcal{J}'}$ with $\mathcal{J}' := \mathcal{J}_{\{1,\ldots,d-1\}} \times \{1, \ldots, r_d\}$ via

$$(20) \qquad \mathcal{M}_{\{d\}}(\mathbf{Y}) = \mathbf{V}^\top \mathcal{M}_{\{d\}}(\mathbf{X}).$$

and use cross approximation to find $\tilde{\mathbf{Y}} \approx \mathbf{Y}$.
Step 3.: Build the final approximation $\tilde{\mathbf{X}}$ from

$$(21) \qquad \mathcal{M}_{\{d\}}(\tilde{\mathbf{X}}) = \mathbf{V}\mathcal{M}_{\{d\}}(\tilde{\mathbf{Y}}).$$

The advantage of applying the cross approximation technique to the tensor $\mathbf{Y}$ instead of $\mathbf{X}$ lies in the reduced size in direction $d = N + 1$ for which we expect $r_d \ll n_\ell$. We now describe in more detail how the three approximation steps are carried out.

In **Step 1**, our aim is to construct an (approximate) basis $\mathbf{V}$ of the column space of $\mathcal{M}_{\{d\}}(\mathbf{X})$. To this end, we use the greedy strategy from Algorithm 1 over a subset $\mathcal{J}_{\text{train}} \subset \mathcal{J}_{\{1,\ldots,d-1\}}$ of column indices. To construct the training set $\mathcal{J}_{\text{train}}$, we use the following

---

Algorithm 1 Find column basis $\mathbf{V}$ of $\mathcal{M}_{\{d\}}(\mathbf{X})$

1: $\mathbf{V} := [\,]$
2: **repeat**
3:    $\boldsymbol{j}^* := \arg\max_{\boldsymbol{j} \in \mathcal{J}_{\text{train}}} \left\| (\mathbf{I} - \mathbf{V}\mathbf{V}^\top)\mathcal{M}_{\{d\}}(\mathbf{X})_{\cdot,\boldsymbol{j}} \right\|_2$
4:    $\mathbf{V} := \text{orth}[\mathbf{V}, \mathcal{M}_{\{d\}}(\mathbf{X})_{\cdot,\boldsymbol{j}^*}]$
5: **until** $\left\| (\mathbf{I} - \mathbf{V}\mathbf{V}^\top)\mathcal{M}_{\{d\}}(\mathbf{X})_{\cdot,\boldsymbol{j}^*} \right\|_2 \le \varepsilon$

---

strategy known from tensor cross approximation [19, Sec. 3.5]. Starting with a random index $\boldsymbol{j} \in \mathcal{J}_{\{1,\ldots,d-1\}}$, we consider the set

$$(22) \qquad \mathcal{J}_{\text{cross}}(\boldsymbol{j}) := \{(j_1, \ldots, j_{i-1}, k, j_{i+1}, \ldots, j_{d-1}) : k \in \mathcal{J}_i,\ i = 1, \ldots, d-1\}$$

which forms a 'cross' with center $\boldsymbol{j}$. Repeating this strategy a few number s of times (say $s = 3$) for random indices $\boldsymbol{j}^1, \ldots, \boldsymbol{j}^s \in \mathcal{J}_{\{1,\ldots,d-1\}}$, we arrive at

$$\mathcal{J}_{\text{train}} := \mathcal{J}_{\text{cross}}(\boldsymbol{j}^1) \cup \ldots \cup \mathcal{J}_{\text{cross}}(\boldsymbol{j}^s),$$

which determines the training set for the first loop of Algorithm 1: In every subsequent loop of Algorithm 1, this set is enriched with s additional (random) crosses. In line 3, we reuse the information computed in the previous loops of the algorithm as much as possible.

Once the matrix $\mathbf{V}$ is constructed, our next aim in **Step 2** is to approximate the tensor $\mathbf{Y} \in \mathbb{R}^{\mathcal{J}'}$ from (20) in the hierarchical tensor format $\mathcal{H}_{\boldsymbol{r}}$. Recalling the main idea of the approach in [3], we seek to recursively approximate the matricizations of $\mathbf{M} = \mathcal{M}_t(\mathbf{Y})$ at any node $t \in T_{\mathcal{D}}$ by a so-called *cross approximation* of the form

$$(23) \qquad\qquad \mathbf{M} \approx \tilde{\mathbf{M}} := \mathbf{M}|_{\mathcal{J}'_t \times \mathcal{C}_t} \cdot \mathbf{M}|_{\mathcal{R}_t \times \mathcal{C}_t}^{-1} \cdot \mathbf{M}|_{\mathcal{R}_t \times \mathcal{J}'_{[t]}}$$

with $\text{rank}(\tilde{\mathbf{M}}) = r_t$ and pivot sets $\mathcal{R}_t \subset \mathcal{J}'_t$, $\mathcal{C}_t \subset \mathcal{J}'_{[t]}$ of size r_t. For each node $t \in T_{\mathcal{D}}$, the rank r_t can be chosen adaptively in order to reach a given (heuristic) target accuracy $\varepsilon_{\text{ten}} \ge 0$ such that $\|\mathbf{M} - \tilde{\mathbf{M}}\|_2 \approx \varepsilon_{\text{ten}}\|\mathbf{M}\|_2$.

The matrices $\mathbf{M}|_{\mathcal{J}'_t \times \mathcal{C}_t}, \mathbf{M}|_{\mathcal{R}_t \times \mathcal{J}'_{[t]}}$ in (23) are never formed explicitly. The essential information for the construction of $\mathbf{Y} \in \mathcal{H}_{\boldsymbol{r}}$ with $\boldsymbol{r} = (r_t)_{t \in T_{\mathcal{D}}}$ are condensed in the pivot sets $\mathcal{R}_t, \mathcal{C}_t$ and the matrices $\mathbf{M}|_{\mathcal{R}_t \times \mathcal{C}_t} \in \mathbb{R}^{r_t \times r_t}$ from (23). This construction is explicit in the sense that the necessary transfer tensors $\mathbf{B}_t$ for all inner nodes $t \in T_{\mathcal{D}}$ and the matrices $\mathbf{U}_t$ in the leaf nodes $t \in T_{\mathcal{D}}$ are directly determined by the values of $\mathbf{Y}$ at certain entries defined by the pivots sets. The details of this procedure can be found in [3, 21].

After the cross approximation has been performed, **Step 3** involves no further approximation but only a simple matrix-matrix product. Assume that the tensor $\mathbf{Y}$ has been approximated by $\tilde{\mathbf{Y}}$ represented in $\mathcal{H}_{\boldsymbol{r}}$ by means of transfer tensors $\mathbf{B}_t$ for inner nodes $t \in T_{\mathcal{D}}$ and matrices $\mathbf{U}_t$ for leaf nodes $t \in T_{\mathcal{D}}$. In the node $t = \{d\}$, we now compute the matrix $\mathbf{U}'_t := \mathbf{V}\mathbf{U}_t$, whereas for all other leaf nodes $t \in T_{\mathcal{D}}$ we keep $\mathbf{U}'_t := \mathbf{U}_t$. It turns out that the tensor $\tilde{\mathbf{X}}$ from (21) is then represented by the transfer tensors $\mathbf{B}_t$ and the matrices $\mathbf{U}'_t$.

5.3. **Error Analysis.** We now study the effect of a perturbed multilevel approximation introduced through tensor approximations $\tilde{\mathbf{X}}^{(\ell)} \approx \mathbf{X}^{(\ell)}$. In particular our aim is to derive an indication from Lemma 8 for the required accuracy in the tensor approximation in order to maintain the convergence result for the multilevel scheme.

Thanks to the orthogonality of the basis $\{\psi_{\ell,i}\}$ in (17), we immediately derive from (18) that

$$\left\|\delta_{N,\ell}(\boldsymbol{y}_{\boldsymbol{k}}^{(\ell)})\right\|_{H_0^1(D)} = \left\|\mathbf{X}_{(\boldsymbol{k},\cdot)}^{(\ell)}\right\|_2, \qquad \boldsymbol{k} \in \mathcal{K}_{\boldsymbol{p}^{(\ell)}}.$$

In order to apply Lemma 8, we need to ensure that

$$\left\|\tilde{\mathbf{X}}_{(\boldsymbol{k},\cdot)}^{(\ell)} - \mathbf{X}_{(\boldsymbol{k},\cdot)}^{(\ell)}\right\|_2 \lesssim 2^{-L}, \qquad \boldsymbol{k} \in \mathcal{K}_{\boldsymbol{p}^{(\ell)}}.$$

Noting that $\left\|\delta_{N,\ell}(\boldsymbol{y}_{\boldsymbol{k}}^{(\ell)})\right\|_{H_0^1(D)} \lesssim 2^{-\ell}$, this can be guaranteed if we require

$$\left\|\tilde{\mathbf{X}}_{(\boldsymbol{k},\cdot)}^{(\ell)} - \mathbf{X}_{(\boldsymbol{k},\cdot)}^{(\ell)}\right\|_2 \lesssim 2^{\ell-L} \left\|\mathbf{X}_{(\boldsymbol{k},\cdot)}^{(\ell)}\right\|_2, \qquad \boldsymbol{k} \in \mathcal{K}_{\boldsymbol{p}^{(\ell)}}.$$

This motivates to perform the tensor approximation with a relative accuracy of $\varepsilon_\ell \sim 2^{\ell-L}$ such that

$$\left\|\tilde{\mathbf{X}}^{(\ell)} - \mathbf{X}^{(\ell)}\right\|_2 \lesssim \varepsilon_\ell \left\|\mathbf{X}^{(\ell)}\right\|_2. \tag{24}$$

As a consequence, the tensor approximation for higher levels ℓ needs to be done less accurate.

5.4. **Final Algorithm.** Compiling all the results obtained so far, our final strategy is summarized in Algorithm 2.

Algorithm 2 Multilevel tensor approximation *ML-Tensor*

1: **for** $\ell = 0, \ldots, L$ **do**
2:　　Choose $h_\ell \sim 2^{-\ell}$, $p^{(\ell)} \sim L - \ell$, $\varepsilon_\ell \sim 2^{\ell-L}$
3:　　Approximate $\tilde{\mathbf{X}}^\ell \approx \mathbf{X}^\ell$ from (18) using Steps 1,2,3 with relative accuracy ε_ℓ
4: **end for**
5: **return** Multilevel approximation $\tilde{u}_L^{\mathrm{ML}}$ from (16)
---

6. Numerical Experiments

In the numerical experiments, we consider the parametric diffusion problem on the unit square given by

$$-\nabla \cdot \big(a(\boldsymbol{y})\nabla u(\boldsymbol{y})\big) = 1, \quad \text{in } D = (0,1)^2,$$
$$u(\boldsymbol{y}) = 0, \quad \text{on } \partial D.$$

On each level ℓ of the proposed multilevel scheme, the domain D is discretized by a uniform triangulation with mesh size

$$h_\ell = 2^{-\ell}h_0, \qquad h_0 = 1/4,$$

using $\mathcal{Q}_1$, i.e., bilinear finite elements with n_ℓ degrees of freedom.

To construct the interpolation operator $\mathcal{I}_{\boldsymbol{p}^{(\ell)}}$ from (5), we use an isotropic polynomial degree on each level defined by

$$\boldsymbol{p}^{(\ell)} = (p^{(\ell)}, \dots, p^{(\ell)}) \in \mathbb{N}_0^N, \qquad p^{(\ell)} := \lfloor (L - \ell + 1)/2 \rfloor.$$

This means that possible anisotropies induced by the decay of the Karhunen-Loève expansion are not considered here. The interpolation points $\boldsymbol{y_k} \in \Gamma = [-1, 1]^N$ are given by the tensorized roots of the Chebyshev polynomials of the first kind of degree $p^{(\ell)} + 1$. The accuracy for the tensor approximation from (24) on each level is chosen as

$$\varepsilon_\ell = 2^{\ell - L} \varepsilon_0, \qquad \varepsilon_0 = 1/4.$$

For each level ℓ, we report the effective rank r_{eff} and the maximal rank $r_{\max}$ of the approximate tensor $\mathbf{X}^{(\ell)}$ represented in the hierarchical tensor format $\mathcal{H}_{\boldsymbol{r}}$. In addition, we state the number of tensor evaluations for **Step 1** and **Step 2** during the cross approximation procedure of the tensor $\mathbf{X}^{(\ell)}$. Note that each evaluation on level ℓ may require the solution of the PDE on level ℓ and level $\ell - 1$.

To measure the interpolation error, we randomly choose $M = 100$ parameters $\boldsymbol{y}^i \in \Gamma$ and compute

$$\varepsilon_L^{\mathrm{ML}}[u] := \left(\sum_{i=1}^M \left\| \tilde{u}_{N,L}^{\mathrm{ML}}(\boldsymbol{y}^i) - u_{N,L}(\boldsymbol{y}^i) \right\|_{H_0^1(D)}^2 \bigg/ \sum_{i=1}^M \left\| u_{N,L}(\boldsymbol{y}^i) \right\|_{H_0^1(D)}^2 \right)^{1/2}.$$

To study the impact of the different levels, we also compute for the perturbed differences $\tilde{\delta}_{N,\ell}$ the error

$$\varepsilon_L^{(\ell)}[u] := \left(\sum_{i=1}^M \left\| \mathcal{I}_{\boldsymbol{p}^{(\ell)}}[\tilde{\delta}_{N,\ell}](\boldsymbol{y}^i) \right\|_{H_0^1(D)}^2 \bigg/ \sum_{i=1}^M \left\| u_{N,L}(\boldsymbol{y}^i) \right\|_{H_0^1(D)}^2 \right)^{1/2}.$$

For a uniform distribution of $y_n \sim \mathcal{U}([-1, 1])$, $n = 1, \dots, N$, we evaluate the expected value of the multilevel solution and compute

$$\varepsilon_L^{\mathbb{E}}[u] := \left\| \mathbb{E}\left[\tilde{u}_{N,L}^{\mathrm{ML}} \right] - \mathbb{E}[u_{\mathrm{ref}}] \right\|_{H_0^1(D)} \bigg/ \left\| \mathbb{E}[u_{\mathrm{ref}}] \right\|_{H_0^1(D)},$$

where u_{ref} is the reference solution obtained from the multilevel scheme on the highest level $L = 7$.

From the multilevel solution, we can immediately compute approximations to output functionals, as, e.g., for

$$\psi(u)] := \int_D u \, \mathrm{d}x.$$

Analogous to the errors for the solution, we then obtain relative errors for the output functional and for the expected value as

$$\varepsilon_L^{\mathbb{E}}[\psi] := \left| \mathbb{E}\left[\psi\left(\tilde{u}_{N,L}^{\mathrm{ML}} \right) \right] - \mathbb{E}[\psi(u_{\mathrm{ref}})] \right| \bigg/ \left| \mathbb{E}[\psi(u_{\mathrm{ref}})] \right|.$$

All numerical experiments have been carried out on a quad-core Intel(R) Xeon(R) CPU E31225 with 3.10GHz. The timings spent on each level ℓ are CPU times for a single core. For the finite element approximation, we have used the software library deal.II, see [5]. All sparse linear systems have been solved by a multifrontal solver from UMFPACK.

6.1. **Karhunen-Loève expansion with exponential decay.** In the first experiment, the Karhunen-Loève expansion of the diffusion coefficient is given by

$$(25) \qquad a(\boldsymbol{y}, x) = 2 + \sum_{n=1}^{N} \sqrt{\lambda_n} b_n(x) y_n$$

with

$$b_n(x) = \sin(2\pi n x_1) \sin(2\pi n x_2).$$

We consider an exponential decay of the eigenvalues defined by $\lambda_n := \exp(-n)$. The results of this experiment for $N = 10, 20$ can be found in Table 1 and Figure 2. From the last

N	ℓ	$p^{(\ell)}$	n_ℓ	r_{eff}	$r_{\max}$	step 1	step 2	time[s]	$\varepsilon_L^{(\ell)}$
10	0	4	25	2.12	4	247	3802	1.7	3.99e-04
	1	3	81	5.48	15	187	7129	11.0	4.24e-04
	2	3	289	11.62	52	559	13839	72.2	6.07e-04
	3	2	1089	14.71	79	262	8011	160.9	1.05e-03
	4	2	4225	13.49	71	340	7562	554.0	6.93e-04
	5	1	16641	7.61	32	225	879	305.5	1.13e-03
	6	1	66049	5.59	20	179	775	1144.8	6.58e-04
	7	0	263169	1.00	1	1	1	11.9	1.92e-03
20	0	4	25	1.89	4	487	12829	7.1	2.71e-04
	1	3	81	4.47	16	367	22680	48.7	4.20e-04
	2	3	289	9.96	56	1099	48864	360.6	4.39e-04
	3	2	1089	12.72	87	422	40206	1135.6	1.08e-03
	4	2	4225	11.74	80	516	35244	3662.6	8.16e-04
	5	1	16641	6.90	39	442	8983	3836.8	1.06e-03
	6	1	66049	4.95	24	316	5870	10698.2	6.93e-04
	7	0	263169	1.00	1	1	1	16.2	1.84e-03

TABLE 1. Karhunen-Loève expansion with exponential decay: Multilevel approximation for $L = 7$ with the number of tensor evaluations for Step 1 and Step 2 and the time spent on each level.

column of Table 1, it can be seen that our adaptive choice of polynomial degree and hierarchical ranks successfully equilibrate the error on the different finite element levels. The hierarchical ranks increase initially and then decrease again as the level increases. This decrease is the most important feature of our approach; it significantly reduces the cost, in terms of queries to the solution, on the finer levels and the overall solution process. Figure 2 shows that the error decreases proportionally with h as the maximum number of levels increases, as expected from our error estimates.

6.2. **Karhunen-Loève expansion with fast algebraic decay.** In this experiment, the diffusion coefficient is again given by (25). We consider an algebraic decay of the eigenvalues defined by $\lambda_n := 1/n^4$. The results of this experiment for $N = 10, 20$ can be found in Table 2 and Figure 3.

6.3. **Karhunen-Loève expansion with slow algebraic decay.** In this experiment, the diffusion coefficient is also given by (25). We consider an algebraic decay of the eigenvalues defined by $\lambda_n := 1/n^2$. The results of this experiment for $N = 10$ can be found in Table 3 and Figure 4. As expected, the maximal hierarchical rank becomes significantly higher compared to the faster algebraic decay.
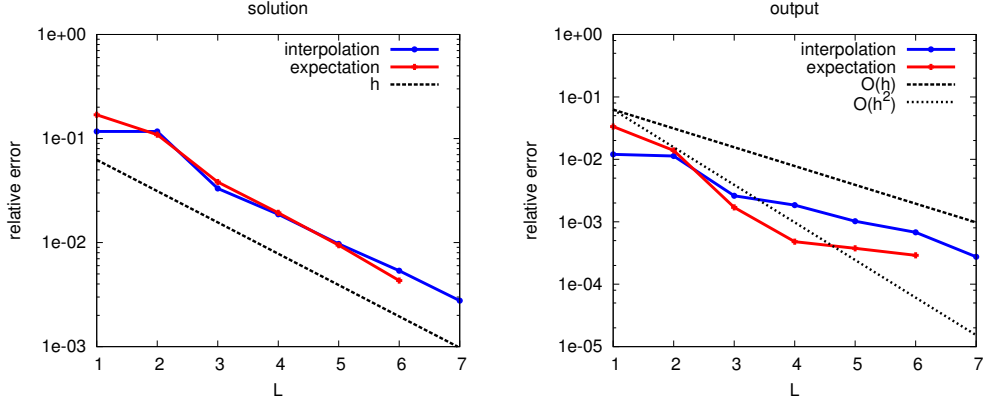
FIGURE 2. Karhunen-Loève expansion with exponential decay for $N = 10$. Left: errors $\varepsilon_L^{\mathrm{ML}}[u]$ and $\varepsilon_L^{\mathbb{E}}[u]$ for the solution. Right: errors $\varepsilon_L^{\mathrm{ML}}[\psi]$ and $\varepsilon_L^{\mathbb{E}}[\psi]$ for the output functional.

N	ℓ	$p^{(\ell)}$	n_ℓ	r_{eff}	r_{max}	step 1	step 2	time[s]	$\varepsilon_L^{(\ell)}$
10	0	4	25	1.84	4	247	2631	1.2	4.20e-04
	1	3	81	4.41	13	280	4492	7.2	4.42e-04
	2	3	289	6.87	28	559	5389	29.7	4.50e-04
	3	2	1089	7.49	31	378	3323	70.9	7.15e-04
	4	2	4225	6.34	23	378	2647	208.1	5.13e-04
	5	1	16641	4.60	14	67	814	239.2	6.98e-04
	6	1	66049	3.62	10	133	732	1004.6	5.53e-04
	7	0	263169	1.00	1	1	1	11.8	1.05e-03
20	0	4	25	1.71	4	487	9458	5.3	3.51e-04
	1	3	81	3.66	14	367	16161	35.0	4.78e-04
	2	3	289	6.17	32	1099	26954	202.6	5.18e-04
	3	2	1089	7.93	46	862	21461	626.7	6.66e-04
	4	2	4225	7.96	43	862	32550	3454.5	4.69e-04
	5	1	16641	5.41	22	316	13526	5634.6	8.01e-04
	6	1	66049	3.65	13	379	6952	12580.2	5.52e-04
	7	0	263169	1.00	1	1	1	16.1	1.16e-03

TABLE 2. Karhunen-Loève expansion with fast algebraic decay: Multilevel approximation for $L = 7$ with the number of tensor evaluations for Step 1 and Step 2 and the time spent on each level

6.4. **Log-uniform case.** Finally, to demonstrate that our approach does not depend on an affine linear decomposition of the diffusion coefficient with respect to the parameters, we consider

$$a(\boldsymbol{y}, x) = \exp\left(\sum_{n=1}^{N} \sqrt{\lambda_n} b_n(x) y_n\right),$$

with an algebraic decay defined by $\lambda_n := 1/n^2$. The results of this experiment for $N = 10$ can be found in Table 4 and Figure 5.
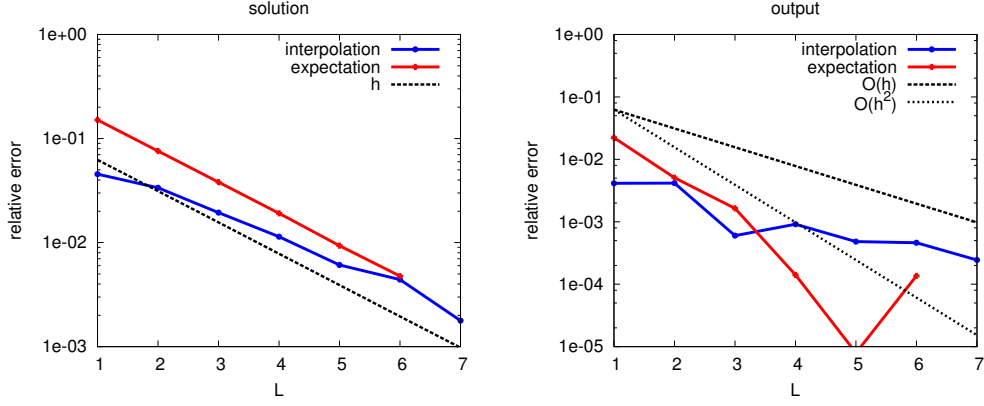
FIGURE 3. Karhunen-Loève expansion with fast algebraic decay for $N = 10$. Left: errors $\varepsilon_L^{\mathrm{ML}}[u]$ and $\varepsilon_L^{\mathbb{E}}[u]$ for the solution. Right: errors $\varepsilon_L^{\mathrm{ML}}[\psi]$ and $\varepsilon_L^{\mathbb{E}}[\psi]$ for the output functional.

N	ℓ	$p^{(\ell)}$	n_ℓ	r_{eff}	$r_{\max}$	step 1	step 2	time[s]	$\varepsilon_L^{(\ell)}$
10	0	4	25	2.36	4	247	4153	1.8	5.87e-04
	1	3	81	5.82	16	187	9050	13.9	4.37e-04
	2	3	289	13.64	58	373	19729	101.3	6.28e-04
	3	2	1089	18.69	95	211	13083	259.8	2.35e-03
	4	2	4225	17.63	93	209	11718	828.7	1.88e-03
	5	1	16641	12.46	67	153	1012	330.0	1.92e-03
	6	1	66049	9.14	41	323	961	1578.5	8.59e-04
	7	0	263169	1.00	1	1	1	12.2	2.51e-03

TABLE 3. Karhunen-Loève expansion with slow algebraic decay: Multilevel approximation for $L = 7$ with the number of tensor evaluations for Step 1 and Step 2 and the time spent on each level
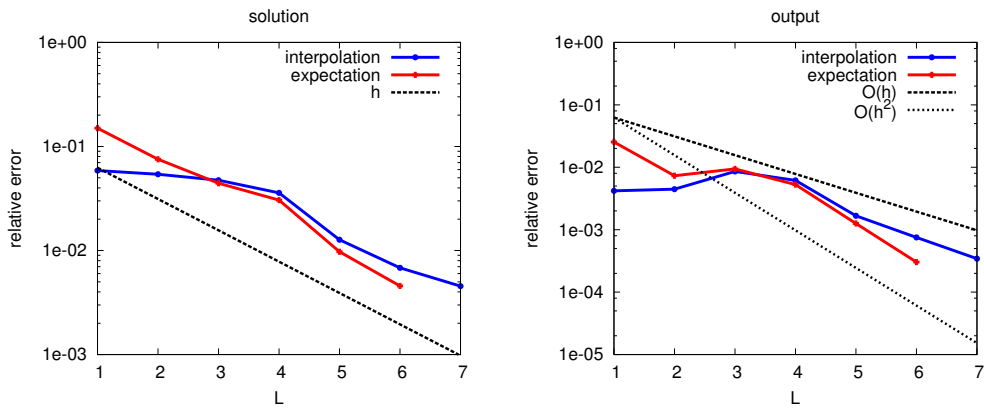

FIGURE 4. Karhunen-Loève expansion with slow algebraic decay for $N = 10$. Left: errors $\varepsilon_L^{\mathrm{ML}}[u]$ and $\varepsilon_L^{\mathbb{E}}[u]$ for the solution. Right: errors $\varepsilon_L^{\mathrm{ML}}[\psi]$ and $\varepsilon_L^{\mathbb{E}}[\psi]$ for the output functional.

N	ℓ	$p^{(\ell)}$	n_ℓ	r_{eff}	$r_{\max}$	step 1	step 2	time[s]	$\varepsilon_L^{(\ell)}$
10	0	4	25	4.85	11	247	31106	13.0	6.40e-04
	1	3	81	9.65	28	187	36430	56.4	7.60e-04
	2	3	289	15.37	60	373	51920	270.3	1.03e-03
	3	2	1089	17.10	90	211	19827	402.4	1.80e-03
	4	2	4225	15.20	80	222	13206	975.7	1.50e-03
	5	1	16641	9.96	46	182	1001	341.0	2.24e-03
	6	1	66049	7.34	28	118	941	1299.9	1.42e-03
	7	0	263169	1.00	1	1	1	12.4	4.64e-03

TABLE 4. Log-uniform case: Multilevel approximation for $L = 7$ with the number of tensor evaluations for Step 1 and Step 2 and the time spent on each level
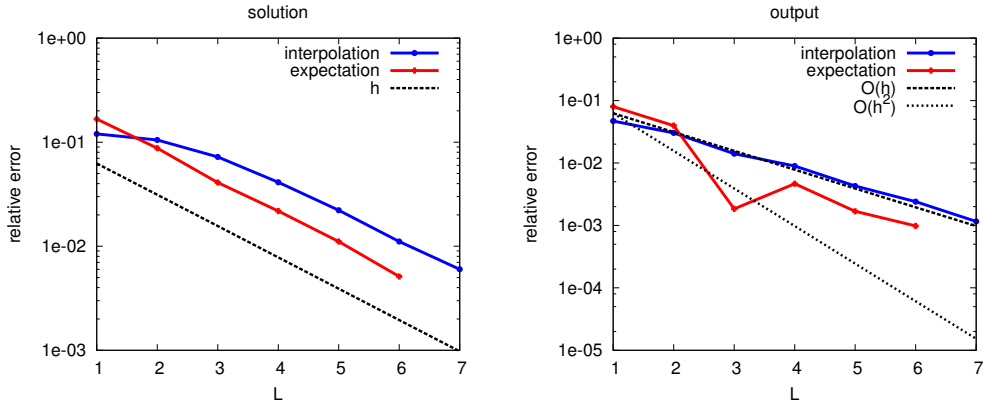

FIGURE 5. Log-uniform case for $N = 10$. Left: errors $\varepsilon_L^{\mathrm{ML}}[u]$ and $\varepsilon_L^{\mathbb{E}}[u]$ for the solution. Right: errors $\varepsilon_L^{\mathrm{ML}}[\psi]$ and $\varepsilon_L^{\mathbb{E}}[\psi]$ for the output functional.

7. Conclusions

In this article, we have considered the multilevel tensor approximation for elliptic partial differential equations with a random diffusion coefficient. By combining the multilevel idea for the approximation in the random parameter, which has firstly been introduced in the context of multilevel Monte Carlo methods, with a hierarchical tensor product approximation, we provide an efficient means to directly represent the solution in a data sparse format. This representation can directly be employed for the evaluation of various functionals of the solution without the necessity of performing additional costly computations. In contrast to previous works, we do not rely on an a priori sparsified representation based on polynomials, but adaptively compute a data sparse representation of the solution with the aid of the hierarchical tensor format and the cross approximation. The numerical results confirm the effectiveness of the presented method.

References

[1] I. Babuška, F. Nobile, and R. Tempone. A stochastic collocation method for elliptic partial differential equations with random input data. *SIAM Rev.*, 52(2):317–355, 2010.

[2] J. Ballani and L. Grasedyck. Hierarchical tensor approximation of output quantities of parameter-dependent PDEs. *SIAM/ASA J. Uncertain. Quantif.*, 3(1):393–416, 2015.

[3] J. Ballani, L. Grasedyck, and M. Kluge. Black box approximation of tensors in hierarchical Tucker format. *Linear Algebra Appl.*, 438(2):639–657, 2013.

[4] J. Ballani and D. Kressner. Reduced basis methods: from low-rank matrices to low-rank tensors. *SIAM J. Sci Comput.*, To appear.

[5] W. Bangerth, R. Hartmann, and G. Kanschat. deal.II – a general purpose object oriented finite element library. *ACM Trans. Math. Softw.*, 33(4):24/1–24/27, 2007.

[6] A. Barth, C. Schwab, and N. Zollinger. Multi-level Monte Carlo finite element method for elliptic PDEs with stochastic coefficients. *Numer. Math.*, 119(1):123–161, 2011.

[7] H.-J. Bungartz and M. Griebel. Sparse grids. *Acta Numer.*, 13:147–269, 2004.

[8] J. Charrier. Strong and weak error estimates for elliptic partial differential equations with random coefficients. *SIAM J. Numer. Anal.*, 50(1):216–246, 2012.

[9] A. Cohen, R. DeVore, and C. Schwab. Convergence rates of best N-term Galerkin approximations for a class of elliptic sPDEs. *Found. Comput. Math.*, 10:615–646, 2010.

[10] S. Dolgov, B. N. Khoromskij, A. Litvinenko, and H. G. Matthies. Polynomial chaos expansion of random coefficients and the solution of stochastic partial differential equations in the tensor train format. *SIAM/ASA J. Uncertain. Quantif.*, 3(1):1109–1135, 2015.

[11] A. Doostan and G. Iaccarino. A least-squares approximation of partial differential equations with high-dimensional random inputs. *J. Comput. Phys.*, 228(12):4332–4345, 2009.

[12] M. Eigel, M. Pfeffer, and R. Schneider. Adaptive stochastic Galerkin FEM with hierarchical tensor representations. Technical report 2015/29, TU Berlin, 2015.

[13] M. Espig, W. Hackbusch, A. Litvinenko, H. G. Matthies, and P. Wähnert. Efficient low-rank approximation of the stochastic Galerkin matrix in tensor formats. *Comput. Math. Appl.*, in press, 2012.

[14] M. Espig, W. Hackbusch, A. Litvinenko, H. G. Matthies, and E. Zander. Efficient analysis of high dimensional data in tensor formats. In *Sparse Grids and Applications*, volume 88 of *Lecture Notes in Computational Science and Engineering*, pages 31–56. Springer, Berlin-Heidelberg, 2013.

[15] L. Fejér. Mechanische Quadraturen mit positiven Cotesschen Zahlen. *Math. Z.*, 37(1):287–309, 1933.

[16] M. Giles. Multilevel Monte Carlo path simulation. *Oper. Res.*, 56(3):607–617, 2008.

[17] M. Giles and B. Waterhouse. Multilevel quasi-Monte Carlo path simulation. *Radon Series Comp. Appl. Math.*, 8:1–18, 2009.

[18] L. Grasedyck. Hierarchical singular value decomposition of tensors. *SIAM J. Matrix Anal. Appl.*, 31:2029–2054, 2010.

[19] L. Grasedyck, D. Kressner, and C. Tobler. A literature survey of low-rank tensor approximation techniques. *GAMM-Mitt.*, 36(1):53–78, 2013.

[20] M. Griebel, H. Harbrecht, and M. Peters. Multilevel quadrature for elliptic parametric partial differential equations on non-nested meshes. *arXiv:1509.09058*, 2015.

[21] W. Hackbusch. *Tensor Spaces and Numerical Tensor Calculus*. Springer, Berlin, 2012.

[22] W. Hackbusch and S. Börm. $\mathcal{H}^2$-matrix approximation of integral operators by interpolation. *Appl. Numer. Math.*, 43(1-2):129–143, 2002.

[23] W. Hackbusch and S. Kühn. A new scheme for the tensor representation. *J. Fourier Anal. Appl.*, 15(5):706–722, 2009.

[24] A.L. Haji Ali, F. Nobile, L. Tamellini, and R. Tempone. Multi-index stochastic collocation for random pdes. *arXiv preprint arXiv:1508.07467*, 2015.

[25] H. Harbrecht, M. Peters, and M. Siebenmorgen. On multilevel quadrature for elliptic stochastic partial differential equations. In J. Garcke and M. Griebel, editors, *Sparse Grids and Applications*, volume 88 of *Lecture Notes in Computational Science and Engineering*, pages 161–179. Springer, Berlin-Heidelberg, 2012.

[26] H. Harbrecht, M. Peters, and M. Siebenmorgen. Multilevel accelerated quadrature for PDEs with log-normally distributed diffusion coefficient. *SIAM/ASA J. Uncertain. Quantif.*, 4(1):520–551, 2016.

[27] S. Heinrich. The multilevel method of dependent tests. In *Advances in stochastic simulation methods (St. Petersburg, 1998)*, Stat. Ind. Technol., pages 47–61. Birkhäuser, Boston, 2000.

[28] S. Heinrich. Multilevel Monte Carlo methods. In *Lecture Notes in Large Scale Scientific Computing*, pages 58–67, London, 2001. Springer.

[29] E. Hille and R. S. Phillips. *Functional Analysis and Semi-Groups*, volume 31. American Mathematical Society, Providence, 1957.

[30] V. A. Hoang and C. Schwab. N-term Wiener chaos approximation rate for elliptic PDEs with lognormal Gaussian random inputs. *Math. Models Methods Appl. Sci.*, 4(24):797–826, 2014.

[31] B. N. Khoromskij and I. Oseledets. Quantics-TT collocation approximation of parameter-dependent and stochastic elliptic PDEs. *Comp. Meth. in Applied Math.*, 10(4):376–394, 2010.

[32] B. N. Khoromskij and C. Schwab. Tensor-structured Galerkin approximation of parametric and stochastic elliptic PDEs. *SIAM J. Sci. Comput.*, 33(1):364–385, 2011.

[33] D. Kressner and C. Tobler. Low-rank tensor Krylov subspace methods for parameterized linear systems. *SIAM J. Matrix Anal. Appl.*, 32(4):1288–1316, 2011.

[34] Frances Y. Kuo, Christoph Schwab, and Ian H. Sloan. Multi-level quasi-Monte Carlo finite element methods for a class of elliptic PDEs with random coefficients. *Found. Comput. Math.*, 15(2):411–449, 2015.

[35] K. Lee and H. C Elman. A preconditioned low-rank projection method with a rank-reduction scheme for stochastic partial differential equations. arXiv:1605.05297, 2016.

[36] H. G. Matthies and E. Zander. Solving stochastic systems with low-rank tensor compression. *Linear Algebra Appl.*, 436(10):3819–3838, 2012.

[37] A. Nouy. Low-rank methods for high-dimensional approximation and model order reduction. *arXiv preprint arXiv:1511.01554*, 2015.

[38] A. Nouy. Low-rank tensor methods for model order reduction. *arXiv preprint arXiv:1511.01555*, 2015.

[39] I. V. Oseledets and E. E. Tyrtyshnikov. TT-cross approximation for multidimensional arrays. *Linear Algebra Appl.*, 432(1):70–88, 2010.

[40] T. J. Rivlin. *The Chebyshev Polynomials*. Wiley, Chichester, 1974.

[41] D. V. Savostyanov and I. V. Oseledets. Fast adaptive interpolation of multi-dimensional arrays in tensor train format. In *Proceedings of 7th International Workshop on Multidimensional Systems (nDS)*. IEEE, 2011.

[42] C. Schwab and C. J. Gittelson. Sparse tensor discretizations of high-dimensional parametric and stochastic PDEs. *Acta Numer.*, 20:291–467, 2011.

[43] C. Schwab and R. Todor. Karhunen-Loève approximation of random fields by generalized fast multipole methods. *J. Comput. Phys.*, 217:100–122, 2006.

[44] A. L. Teckentrup, P. Jantsch, C. G. Webster, and M. Gunzburger. A multilevel stochastic collocation method for partial differential equations with random input data. *SIAM/ASA J. Uncertain. Quantif.*, 3(1):1046–1074, 2015.

[45] C. Tobler. *Low-rank Tensor Methods for Linear Systems and Eigenvalue Problems*. PhD thesis, ETH Zürich, 2012.

JONAS BALLANI, DANIEL KRESSNER, MATHICSE-ANCHP, ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE, STATION 8, CH-1015 LAUSANNE.,

E-mail address: {jonas.ballani,daniel.kressner}@epfl.ch

MICHAEL PETERS, DEPARTMENT OF MATHEMATICS AND COMPUTER SCIENCE, UNIVERSITY OF BASEL, SPIEGELGASSE 1, CH-4051 BASEL,

E-mail address: michael.peters@unibas.ch

LATEST PREPRINTS

No.	**Author:** *Title*

2015-31 **T. Boulenger, D. Himmelsbach, E. Lenzmann**
Blowup for fractional NLS

2015-32 **A. Hyder**
Moser functions and fractional Moser-Trudinger type inequalities

2015-33 **S. Zimmermann**
The Cremona group of the plane is compactly presented

2015-34 **S. Zimmermann**
The Abelianisation of the real Cremona group

2015-35 **J. Dölz, H. Harbrecht, M. Peters**
H-matrix based second moment analysis for rough random fields and finite element discretizations

2015-36 **J. K. Canci**
Good reduction for endomorphisms of the projective line in terms of the branch locus

2015-37 **J. K. Canci, L. Paladino**
Preperiodic points for rational functions defined over a global field in terms of good reduction

2015-38 **J. K. Canci**
Preperiodic points for rational functions defined over a rational function field of characteristic zero

2015-39 **J. K. Canci, S. Vishkautsan**
Quadratic maps with a periodic critical point of period 2

2015-40 **H. Harbrecht, M. Peters**
The second order perturbation approach for PDEs on random domains

2015-41 **H. Harbrecht, W. L. Wendland, N. Zorii**
Minimal energy problems for strongly singular Riesz kernels

2015-42 **H. Harbrecht, M. Utzinger**
On adaptive wavelet boundary element methods

2016-01 **S. Iula, G. Mancini**
Extremal functions for singular Moser-Trudinger embeddings

Preprints are available under *https://math.unibas.ch/research/publications*

LATEST PREPRINTS

No.	**Author:** *Title*

2016-02   **J. K. Canci, L. Paladino**
On preperiodic points of rational functions defined over $\mathbb{F}_p(t)$

2016-03   **H. Harbrecht, R. Schneider**
A note on multilevel based error estimation

2016-04   **C. Urech**
On homomorphisms between Cremona groups

2016-05   **A. P. Choudhury, H. Heck**
Stability of the inverse boundary value problem for the biharmonic operator:
Logarithmic estimates

2016-06   **M. Dambrine, I. Greff, H. Harbrecht, B. Puig**
Numerical solution of the homogeneous Neumann boundary value problem
on domains with a thin layer of random thickness

2016-07   **G. Alberti, G. Crippa, A. L. Mazzucato**
Exponential self-similar mixing by incompressible flows

2016-08   **M. Bainbridge, P. Habegger, M. Möller**
Teichmüller curves in genus three and just likely intersections in $G_m^n \times G_a^n$

2016-09   **Gabriel A. Dill**
Effective approximation and Diophantine applications

2016-10   **J. Blanc, S. Zimmermann**
Topological simplicity of the Cremona groups

2016-11   **I. Hedén, S. Zimmermann**
The decomposition group of a line in the plane

2016-12   **J. Ballani, D. Kressner, M. Peters**
Multilevel tensor approximation of PDEs with random data

Preprints are available under *https://math.unibas.ch/research/publications*