Adaptive non-intrusive reconstruction of solutions to high-dimensional parametric PDEs

Martin Eigel[1], Nando Farchmin[2], Sebastian Heidenreich[2], Philipp Trunschke[3]

submitted: December 2, 2021

[1] Weierstrass Institute
     Mohrenstr. 39
     10117 Berlin
     Germany
     E-Mail: martin.eigel@wias-berlin.de

[2] Physikalisch Technische Bundesanstalt
     Abbestr. 2-12
     10587 Berlin
     Germany
     E-Mail: nando.farchmin@ptb.de
             sebastian.heidenreich@ptb.de

[3] Technische Universität Berlin
     Institut für Mathematik
     Straße des 17. Juni 136
     10623 Berlin
     Germany
     E-Mail: trunschke@math.tu-berlin.de

Adaptive non-intrusive reconstruction of solutions to high-dimensional parametric PDEs

Martin Eigel, Nando Farchmin, Sebastian Heidenreich, Philipp Trunschke

ABSTRACT. Numerical methods for random parametric PDEs can greatly benefit from adaptive refinement schemes, in particular when functional approximations are computed as in stochastic Galerkin and stochastic collocations methods. This work is concerned with a non-intrusive generalization of the adaptive Galerkin FEM with residual based error estimation. It combines the non-intrusive character of a randomized least-squares method with the a posteriori error analysis of stochastic Galerkin methods. The proposed approach uses the Variational Monte Carlo method to obtain a quasi-optimal low-rank approximation of the Galerkin projection in a highly efficient hierarchical tensor format. We derive an adaptive refinement algorithm which is steered by a reliable error estimator. Opposite to stochastic Galerkin methods, the approach is easily applicable to a wide range of problems, enabling a fully automated adjustment of all discretization parameters. Benchmark examples with affine and (unbounded) lognormal coefficient fields illustrate the performance of the non-intrusive adaptive algorithm, showing best-in-class performance.

1. INTRODUCTION

High-dimensional parametric *partial differential equations* (PDEs) play a crucial role in modern simulation methods used in the natural sciences and engineering. Especially when uncertainties or variations in the data should be incorporated into the physical model, a parameter vector determining the data realizations leads to a discretization complexity that easily becomes extremely challenging to tackle due to the inherent "curse of dimensionality". There has been very active research activity in particular in the area of Uncertainty Quantification (UQ) to better understand the structure of the problem and to mitigate the numerical obstacles with new methods. Apart from sampling methods for the estimation of quantities of interest such as Monte Carlo sampling, functional approximations allow to exploit commonly encountered structured regularity of the PDE solutions to obtain much higher convergence rates. The central contribution of this paper is the development of an adaptive sample-based Galerkin method in low-rank tensor format, which can be considered a non-intrusive generalization of the adaptive stochastic Galerkin FEM (ASGFEM) as e.g. derived in [1–3]. It combines and generalizes previous results on the residual based reliable error estimator with the non-intrusive low-rank tensor reconstruction techniques from [4, 5] and [6]. In contrast to the frequently used intrusive ASGFEM, the presented method is a versatile generalization that could easily be applied to a broad range of problems with only small modifications. A main feature it shares with stochastic collocation (SC) methods is its sole dependence on pointwise solutions. However, opposite to SC, a hierarchical tensor compression of the solution and coefficient field usually lead to a beneficial scaling with respect to the parameter dimensions, allowing to compute very high-dimensional problems that might otherwise only be tractable by neural network representations.

As a model problem we consider the parameter dependent Darcy equation

$$\begin{aligned}
-\operatorname{div}_x a(x,y)\,\nabla_x u(x,y) &= f(x) &&\text{in } D, \\
u(x,y) &= 0 &&\text{on } \partial D,
\end{aligned}$$

(1.1)

on some domain $D \subset \mathbb{R}^2$, where $y = (y_1, \ldots, y_L) \subset \mathbb{R}^L$ is a high or even infinite dimensional parameter vector determining the (affine or nonlinear in y) coefficient field and hence the solution. To counter the resulting (possibly) extensive memory complexity caused by a functional representation, the coefficient tensors of $a(x,y)$ and $u(x,y)$ are low-rank approximated in the *Tensor Train* (TT) format via a least-squares regression technique coined the *Variational Monte Carlo* (VMC) method [4]. Given a sufficient amount of training samples, which can be estimated by a heuristic criterion, the

obtained approximation is equivalent to the stochastic Galerkin projection with high probability [5, 7]. In comparison to Monte Carlo methods, exploiting the regularity and low-rank approximability of the solution may drastically improve the rate of convergence as shown in [4].

The resulting adaptive algorithm only requires access to samples $u(y^{(i)})$ (and possibly $a(y^{(i)})$) generated via a black-box solver. Given an approximate solution reconstruction $w_N \approx u$, contributions of the deterministic, stochastic and algebraic erros can be evaluated, leading to a reliable upper bound of the energy error of the form

$$(1.2) \qquad \|u - w_N\|_B \lesssim \eta_{\det}(w_N) + \eta_{\mathrm{sto}}(w_N) + \eta_{\mathrm{alg}}(w_N).$$

These error estimates then steer the local refinements of the discrete space. It is noteworthy that no direct interaction with the parametric solver is required besides communicating how the underlying mesh has to be changed. The error estimators can be proven to be reliable and efficient in many cases, see [1, 8–10] and [11] for a review of the underlying deterministic derivation.

We demonstrate the performance of the proposed algorithm for the model problem (1.1) with both affine and lognormal diffusion coefficients $a(x, y)$. It should be pointed out that our algorithm can be applied with only minor modifications to any linear or (mild) nonlinear problem that permits computable a posteriori error bounds.

Structure. Section 2 introduces the model problem setting, its variational formulation as well as the spatial and stochastic discretization. Section 3 then examines how the diffusion field and the solution of (1.1) can be obtained efficiently and non-intrusively. In Section 4, we recall the residual based error estimator from [1, 3] and derive a heuristic for controlling the regression error. The resulting adaptive refinement strategy and the overall algorithm are presented in Section 5. Finally, we test the fully adaptive scheme in several examples with affine and lognormal coefficient fields in Section 6.

Related work. Theoretical considerations about the class of parametric PDEs used in this paper (1.1) can e.g. be found in the review articles [12, 13]. Moreover, the technically involved lognormal case is analysed in detail in [14–16].

Functional representations of high-dimensional problem solutions can be obtained by spectral approximations, enabling optimal convergence rates numerically. Most prominent are SC [17–19] and SG [20, 21] methods. SC has the advantage of being non-intrusive and thus is easy to use with already existing simulation codes, whereas SG methods can be understood as an extension of classical finite element (FE) methods, requiring a problem-specific implementation. To make these methods computationally feasible, different model reduction techniques can be incorporated. In the context of this paper, two are most relevant: (i) adaptivity based on computable error estimators and (ii) low-rank compression with hierarchical tensor formats.

Adaptive algorithms based on a posteriori estimators have been developed to increase the physical FE space and the stochastic space automatically and problem dependent. They can be seen as an extension of adaptive methods in deterministic FEM, see [1, 8, 22] for residual based estimators and [23–26] for hierarchical estimators. Alternative adjoint approaches for quantities of interest can be found in [27, 28].

Refinement of the stochastic space typically incorporates enlarging the global polynomial basis by increasing the polynomial degrees and including more stochastic modes. When using tensor formats, an additional error contribution that has to be controlled is the algebraic (compression) error, leading to an adjustment of the representation rank. In case of affine coefficient fields, there are first results on the convergence of adaptive algorithms [8, 9] and even optimality [10] under certain conditions. Recent results provide optimality in much greater generality by using wavelet expansions [29].

To circumvent exponential growth of the stochastic discretization space, hierarchical tensor formats can be used if the problem is low-rank representable. Some details on different tensor formats and

numerical algorithms can be found in [30–33]. Our focus lies on the TT format [33], which has been used with tremendous success for the solution of parametric PDEs and related UQ problems such as Bayesian inversion and random field representations, see [34–38]. ASGFEM in hierachical tensor formats are presented in [2, 39] for affine coefficients. The first ASGFEM for lognormal coefficients is developed in [3]. In contrast to affine fields, nonlinear expansions as in the lognormal case cannot be represented easily in tensor formats and are in fact rather challenging to obtain. Accordingly, a limitation of the ASGFEM in [3] is that it relies on a specifically tailored construction of the parametric field. Other approaches using some tensor completion can be found in [35, 40]. In [6], a general projection based method is developed to govern the approximation error of arbitrary surrogates. Since it can be employed with a wide range of coefficient fields, it is very much in the spirit of our method and is used in Section 3. The basis for the least-squares tensor regression of parametric PDE solutions was laid with the VMC method in [4]. An alternative technique is the tensor cross approximation of [41]. From a practical point of view, a major difference of the two methods lies in the integration of training samples. The cross approximation evaluates the parametric black-box solver during runtime as the interpolation points are chosen adaptively ("active learning"), whereas the VMC method assumes precomputed (randomly sampled) evaluations of the solver for random parameter realizations ("passive learning").

2. THE MODEL PROBLEM

This section establishes the analytical foundations of the model problem (1.1). We recall some details on the functional setting for two common types of random coefficient fields and point to references for an in depth analysis when necessary. Throughout this work we assume $D \subset \mathbb{R}^2$ to be a polygonal bounded Lipschitz domain. Moreover, without loss of generality, we limit ourselves to a deterministic source term $f \in L^2(D)$ and homogeneous Dirichlet boundary conditions since modelling the right-hand side and the boundary conditions as stochastic fields independent on the diffusion coefficient $a(x, y)$ would not introduce significant modifications. With typical applications in e.g. stochastic groundwater flow modelling, the diffusion coefficient is often defined by a Karhunen-Loève type expansion of the form

$$(2.1) \qquad \gamma(x, y) = \gamma_0(x) + \sum_{\ell=1}^{L} \gamma_\ell(x) y_\ell \qquad \text{for } x \in D,$$

and almost all $y_\ell \sim \pi_\ell$ with independent distributions $\pi_\ell, L \in \mathbb{N} \cup \{\infty\}$. In many applications, however, the far more challenging exponential diffusion field

$$(2.2) \qquad \kappa(x, y) = \exp(\gamma(x, y) - \gamma_0(x, y))$$

has to be considered. Solvability of (1.1) for $a = \gamma$ follows directly from the uniform boundedness and positivity of the affine field (2.1) [12]. Well-posedness of (1.1) for the unbounded case $a = \kappa$ is significantly more involved and requires the introduction of adapted function spaces, cf. [14, 16, 42–44]. We refer to [3] for a concise review of the concepts that we use for the problem setting (1.1).

Let $\mathcal{X} := H_0^1(D)$ equipped with the standard norm $\|w\|_{\mathcal{X}} = \|\nabla w\|_{L^2(D)}$ and let $\mathcal{F} := \{\mu \in \mathbb{N}_0^\infty : |\operatorname{supp} \mu| < \infty\}$ be the set of finitely supported multi-indices, where $\operatorname{supp} \mu$ denotes the set of all indices of μ different from zero. For any $m \in \mathbb{N}_0$ and $n \in \mathbb{N}$, let $[m:n] := \{m, \dots, n-1\}$, where $[m:n] := \{0\}$ if $m \geqslant n$ and $[n] := [0:n]$. Define the full tensor index set

$$(2.3) \qquad \Lambda_d := [d_1] \times \dots [d_L] \times [1] \times \cdots \subset \mathcal{F}, \qquad \text{for } L \in \mathbb{N} \text{ and } d \in \mathbb{N}^L.$$

By $\{P_j^\ell\}_{j=0}^\infty$ we denote a set of orthogonal and normalized polynomials in $L^2(\Gamma_\ell, \pi_\ell)$, where we assume $\Gamma_\ell \subset \mathbb{R}$ for $\ell \in \mathbb{N}$. Moreover, we consider the tensor product case $\Gamma = \prod_{\ell \in \mathbb{N}} \Gamma_\ell$ and $\pi = \prod_{\ell \in \mathbb{N}} \pi_\ell$ and define an orthonormal product basis $\{P_\mu\}_{\mu \in \mathcal{F}}$ of $L^2(\Gamma, \pi)$ by $P_\mu(y) := \prod_{\ell=1}^\infty P_{\mu_\ell}^\ell(y_\ell) =$

$\prod_{\ell\in\mathrm{supp}\,\mu} P^\ell_{\mu_\ell}(y_\ell)$. Note that the use of global polynomials is justified by the high regularity of the solution of (1.1) with respect to the random variables [42, 45, 46]. In our numerical experiments we rely on (scaled) Hermite and Legendre polynomials as univariate basis functions. Details on the normalization constants for the respective polynomials and an analytical expression for the triple products $\tau_{ijk} = \mathbb{E}[P^\ell_i P^\ell_j P^\ell_k]$ are given in Supplement A. We define the bilinear form

$$(2.4) \qquad B(w,v) := \int_\Gamma \int_D a(x,y)\,\nabla\,w(x,y)\cdot\nabla\,v(x,y)\,\mathrm{d}x\,\mathrm{d}\pi(y)$$

on $L^2(\Gamma,\pi;\mathcal{X})$ and denote the induced energy norm by $\|w\|_B := B(w,w)^{1/2}$. We additionally abbreviate $\|w\|_{\pi,D} = \|w\|_{L^2(\Gamma,\pi;L^2(D))}$. The variational form of (1.1) then reads

$$(2.5) \qquad B(u,v) = F(v) \quad \text{for all } v \in \mathcal{V},$$

where $F(v) = \int_\Gamma \int_D f(x)v(x,y)\,\mathrm{d}x\,\mathrm{d}\pi(y)$ is supposed to be well defined for an appropriate Hilbert space $\mathcal{V}$.

A conforming FE space $\mathcal{X}_p(\mathcal{T}) := \mathrm{span}\{\varphi_i\}_{j=1}^N$ is used as spatial discretization of the physical space $\mathcal{X}$. In particular, we assume $\mathcal{T}$ to be a regular triangulation of the domain D with edges $\mathcal{E}$ and consider for $p \in \mathbb{N}$ the standard conforming order-p Lagrange elements such that $\mathcal{X}_p(\mathcal{T}) := P_p(\mathcal{T}) \cap C(\bar{\mathcal{T}})$, where $P_p(\mathcal{T})$ is the space of element-wise polynomials of order p. For any element $T \in \mathcal{T}$ and edge $E \in \mathcal{E}$, let h_T and h_E denote the diameter of T and E, respectively. Define the normal jump of a function $w \in H^1(D;\mathbb{R}^2)$ over the edge $E = \bar{T}_1 \cap \bar{T}_2$ by $[\![w]\!]_E = (w|_{T_1} - w|_{T_2}) \cdot \nu_E$ for the edge normal vector $\nu_E = \nu_{T_1} = -\nu_{T_2}$ of E. Since the direction of the normal ν_E depends on the enumeration of the neighbouring triangles, we assume an arbitrary but fixed choice of the sign of ν_E for each $E \in \mathcal{E}$. This allows us to define the fully discrete approximation space by

$$(2.6) \qquad \mathcal{V}_N := \mathcal{V}_N(\Lambda_d;\mathcal{T},p) := \Big\{v_N = \sum_{\mu\in\Lambda_d} v_{N,\mu}P_\mu \text{ with } v_{N,\mu} \in \mathcal{X}_p(\mathcal{T}) \text{ for all } \mu \in \Lambda_d\Big\} \subset \mathcal{V}.$$

As a consequence, the Galerkin projection $u_N \in \mathcal{V}_N$ of the solution u of (2.5) is determined uniquely by

$$(2.7) \qquad B(u_N,v_N) = F(v_N) \qquad \text{for all } v_N \in \mathcal{V}_N.$$

To describe the lognormal case $a = \kappa$, define the set of admissible parameters by

$$(2.8) \qquad \Gamma_\kappa := \{y \in \mathbb{R}^L \colon \sum_{\ell=1}^L \|\gamma_\ell\|_{L^\infty(D)}|y_\ell| < \infty\}.$$

Note that for $\sigma_\ell > 0$ the probability density function for the univariate Gaussian distribution $\mathcal{N}(0,\sigma_\ell^2)$ can be written as

$$\pi_\ell(y_\ell;\sigma_\ell) = \zeta_\ell(y_\ell;\sigma_\ell)\frac{1}{\sqrt{2\pi}}\exp(-\frac{1}{2}y_\ell^2) \quad \text{with} \quad \zeta_\ell(y_\ell;\sigma_\ell) := \frac{1}{\sigma_\ell}\exp\Big(\big(\frac{1}{2} - \frac{1}{2\sigma_\ell^2}\big)y_\ell^2\Big).$$

Moreover, for any $\rho \geqslant 0$ let $\sigma_\ell(\rho) := \exp(\rho\|\gamma_\ell\|_{L^\infty(D)})$ and let

$$(2.9) \qquad \zeta_\rho(y) := \prod_{\ell=1}^L \zeta_\ell(y_\ell;\sigma(\rho)) \qquad \text{and} \qquad \pi_\rho := \prod_{\ell=1}^L \pi_\ell(y_\ell;\sigma(\rho)).$$

Note that $\rho = 0$ implies $\zeta_0 = 1$ and thus π_0 denotes the standard Gaussian density. We henceforth investigate the following two application cases.

(A) **affine case.** Consider the affine coefficient field $a(y) = \gamma(y) \in L^\infty(D)$. In this setting, we assume the univariate random variables y_ℓ to be i.i.d. uniformly distributed on $\Gamma_\ell = [-1,1]$, i.e. $\pi_\ell = 1/2$. Consequently, we set $\mathcal{V} = L^2(\Gamma,\pi;\mathcal{X})$ and employ Legendre polynomials as basis functions. For notational convenience, we additionally set $\zeta_\rho(y) \equiv 1$ and $\pi_\rho \equiv \pi = 2^{-L}$ for any $\rho \in \mathbb{R}$ for the analysis later on.

(B) **lognormal case.** Consider the exponential coefficient field $a(y) = \kappa(y) \in L^\infty(D)$. Since we assume the univariate random variables y_ℓ to follow an i.i.d. standard normal distribution, we refer to κ as a lognormal coefficient field. For $\vartheta \in [0, 1]$, $\rho > 0$ and $\Gamma = \Gamma_\kappa$, the solution space is defined by

$$\mathcal{V} = \{w \colon \Gamma \to \mathcal{X} \text{ measurable with } B(w, w) < \infty\},$$

where B is the bilinear form (2.4) with $\pi = \pi_{\vartheta\rho}$ from (2.9). For the polynomial basis we choose scaled Hermite polynomials $\{H_\mu^{\vartheta\rho}\}_{\mu \in \mathcal{F}}$, see Supplement A and [12].

3. DISCRETIZATION OF SOLUTION AND COEFFICIENT

This section describes the approximation of the discrete solution u_N of (2.7), the right-hand side f and the coefficient fields γ and κ in the TT format. A brief summary of notation and some fundamental properties of the TT format are given in Appendix C. A general and more detailed description of the TT format is given in [33, 47, 48] and in e.g. [3, 30, 35, 49] TT representations have been applied to the elliptic model problem (1.1).

3.1. **TT approximation of the solution.** In the following we recall the notion of nonlinear least-squares approximation and show that a sample-based quasi-best approximation of u can be obtained with high probability given sufficiently many samples. For this, recall that the discrete solution u_N of (2.7) satisfies the Galerkin orthogonality property $B(u_N - u, v_N) = 0$ for all $v_N \in \mathcal{V}_N$, which implies that $\|u - v_N\|_B = \|u - u_N\|_B + \|u_N - v_N\|_B$ for any $v_N \in \mathcal{V}_N$. This means that u_N is the $\| \bullet \|_B$-best approximation to the solution u in $\mathcal{V}_N$. Extending this idea to the subset $\mathcal{M}_r = \{w_N \in \mathcal{V}_N \colon \text{tt-rank}(\boldsymbol{w}) \leqslant r\}$ leads to the best approximation problem

$$\underset{w_N \in \mathcal{M}_r}{\arg\min} \|u - w_N\|_B = \underset{w_N \in \mathcal{M}_r}{\arg\min} \|u_N - w_N\|_B.$$

Minimizing the energy norm is straight-forward for the affine case (A), since $\mathcal{V} = L^2(\Gamma, \pi; \mathcal{X})$. For the lognormal case (B) however, minimizing the $\| \bullet \|_B$-norm introduces an additional dependence on the diffusion coefficient. Even though this is not problematic from a theoretical point of few, it is possible to eliminate this dependence by employing the boundedness of the bilinear form B (B.1), i.e.

$$\|u_N - w_N\|_B \leqslant \sqrt{\hat{c}(\vartheta\rho)}\|u_N - w_N\|_{L^2(\Gamma, \pi_\rho; \mathcal{X})}.$$

We then aim to find the minimum

$$(3.1) \qquad u_{N,r} := \underset{w_N \in \mathcal{M}_r}{\arg\min} \|u_N - w_N\|_{L^2(\Gamma, \pi_\rho; \mathcal{X})},$$

which has the advantage of being independent of the choice of ϑ. Since computing the L^2-norm with respect to π_ρ is infeasible in practice, we follow the ideas of [4, 50] and replace the high-dimensional integral with the Monte Carlo estimate

$$(3.2) \qquad \|v\|_n := \sqrt{\frac{1}{n}\sum_{i=1}^n \|v(y^{(i)})\|_{\mathcal{X}}^2}$$

for any $v \in \mathcal{V}$, where the samples $y^{(i)} \sim \pi_\rho$ are independent for all $i = 1, \dots, n$. The computation of the best approximation then reads

$$(3.3) \qquad u_{N,r,n} := \underset{w_N \in \mathcal{M}_r}{\arg\min} \|u_N - w_N\|_n,$$

where the $u_N(y_i)$ can be computed with an arbitrary FE solver. The resulting *nonlinear least squares* problem (3.3) is easy to implement and many highly optimized frameworks exist to solve this problem [50]. Note that this is a fully non-intrusiven approach since it only requires the (pointwise) standard FE solutions $u_N(y^{(i)})$ of the deterministic problem (1.1).

It remains to show is that the minimizer of (3.3) is comparable to the best-approximation $u_{N,r}$ of (3.1) given a sufficiently large number of samples n. Let $\tilde{\mathcal{V}} := L^2(\Gamma, \pi_\rho; \mathcal{X})$ and let $\delta > 0$. We say that the *restricted isometry property* $\mathrm{RIP}_{\mathcal{M}_r}(\delta)$ holds if

$$(3.4) \qquad (1 - \delta)\|w_N\|_{\tilde{\mathcal{V}}}^2 \leqslant \|w_N\|_n^2 \leqslant (1 + \delta)\|w_N\|_{\tilde{\mathcal{V}}}^2 \qquad \text{for all } w_N \in \mathcal{M}_r.$$

Applying the following theorem from [5] guarantees that the minimizer $u_{N,r,n}$ is a quasi-best approximation in $\mathcal{M}_r$.

Theorem 3.1. *Assume that* $\mathrm{RIP}_{\{u_{N,r}\}-\mathcal{M}_r}(\delta)$ *holds. Then*

$$\|u_{N,r} - u_{N,r,n}\|_{\tilde{\mathcal{V}}} \leqslant 2\frac{1}{\sqrt{1-\delta}}\|u - u_{N,r}\|_\infty,$$

where $\|w\|_\infty^2 := \mathrm{ess\,sup}_{y\in\Gamma} \|w(y)\|_{\mathcal{X}}^2$. *If in addition* $\mathrm{RIP}_{\{u-u_{N,r}\}}(\delta)$ *is satisfied, then*

$$\|u_{N,r} - u_{N,r,n}\|_{\tilde{\mathcal{V}}} \leqslant 2\frac{\sqrt{1+\delta}}{\sqrt{1-\delta}}\|u - u_{N,r}\|_{\tilde{\mathcal{V}}}$$

and consequently

$$\|u - u_{N,r}\|_{\tilde{\mathcal{V}}} \leqslant \|u - u_{N,r,n}\|_{\tilde{\mathcal{V}}} \leqslant \left(1 + 2\frac{\sqrt{1+\delta}}{\sqrt{1-\delta}}\right)\|u - u_{N,r}\|_{\tilde{\mathcal{V}}}.$$

Moreover, in [5] a qualitative bound for $n(r, \delta, p)$ is derived such that both RIPs in Theorem 3.1 hold with probability $1 - p$ if $n \geqslant n(r, \delta, p)$.

Remark 3.2. *The norm equivalence* (3.4) *holds strictly speaking only for the affine case (A) in the described form. For the lognormal case (B), an adapted sampling density (cf. [5, 51]) is required. Nevertheless, in practice we do not observe that this is necessary. Moreover, we note that the sampling bound* $n(r, \delta, p)$ *established in [5] is a worst-case bound and that a significantly smaller number of samples suffices in our experiments.*

3.2. **TT representation of the diffusion coefficient.**

Affine coefficient field and right-hand side. The constant right-hand side f and the affine coefficient field γ can be understood as an expansion in univariate multidimensional polynomials. Such functions possess a natural (exact) representation of their coefficient tensor in the TT format. For $d \in \mathbb{N}^L$, consider the set of univariate L-dimensional indices

$$\Delta_d := \bigcup_{\ell=1}^{L}\{j\,e_\ell\colon j \in [d_\ell]\} \qquad \text{with} \qquad |\Delta_d| = 1 + \sum_{\ell=1}^{L}(d_\ell - 1),$$

where e_ℓ denotes the unit vector $(e_\ell)_j = \delta_{\ell j}$ for $j \in \mathbb{N}_0$, and let $\iota\colon \{0, \dots, |\Delta_d| - 1\} \to \Delta_d$ with $\iota(0) = (0, \dots, 0)$ be an arbitrary enumeration of Δ_d. Furthermore, let $\iota_{\mu_\ell} = \iota^{-1}(\mu_\ell e_\ell)$ be the enumeration index of $\mu_\ell e_\ell$ and define for $k \in \mathbb{N}_0$

$$\delta(k, \mu_\ell) := \delta_{k0}\delta_{\mu_\ell 0} + (1 - \delta_{k0})\big(\delta_{\mu_\ell 0} + \delta_{k\iota_{\mu_\ell}}(1 - \delta_{\mu_\ell 0})\big).$$

Any function $w_N \in \mathcal{V}_N(\Delta_d; \mathcal{T}, p)$ can formally be written as an expansion with respect to the full tensor set Λ_d,

$$(3.5) \qquad w_N(x, y) = \sum_{t=0}^{|\Delta_d|-1} w_{\iota(t)}(x)P_{\iota(t)}(y) = \sum_{j\in[|\mathcal{T}|]}\sum_{\mu\in\Lambda_d} \boldsymbol{w}[j, \mu]\varphi_j(x)P_\mu(y),$$

where $\boldsymbol{w}[j, \mu] = 0$ for $\mu \notin \Delta_d$. The following proposition shows that the coefficient tensor $\boldsymbol{w}$ of (3.5) has an exact representation in the TT format.

Proposition 3.3. *For any $w_N \in \mathcal{V}_N(\Delta_d; \mathcal{T}, p)$ the coefficient tensor $\boldsymbol{w} \in \mathbb{R}^{N \times d}$ has an exact representation in the TT format. This representation is given by*

$$\boldsymbol{w}[j, \mu] = \sum_{k_1=0}^{|\Delta_d|-1} \cdots \sum_{k_M=0}^{|\Delta_d|-1} \boldsymbol{w}_0[j, k_1] \prod_{m=1}^{M} \boldsymbol{w}_m[k_m, \mu_m, k_{m+1}],$$

with spatial component tensor $\boldsymbol{w}_0 \in \mathbb{R}^{N \times |\Delta_d|}$ given by the FE coefficients of the functions $w_{\iota(t)}(x)$, i.e.

$$w_{\iota(t)}(x) := \sum_{j \in [|\mathcal{T}|]} \boldsymbol{w}_0[j, t] \varphi_j(x) \qquad \text{for all } t = 0, \dots, |\Delta_d| - 1,$$

and stochastic cores $\boldsymbol{w}_\ell \in \mathbb{R}^{|\Delta_d| \times d_\ell \times |\Delta_d|}$ for $\ell = 1, \dots, L-1$ and $\boldsymbol{w}_L \in \mathbb{R}^{|\Delta_d| \times d_L}$ given by

$$\boldsymbol{w}_\ell[k_\ell, \mu_\ell, k_{\ell+1}] := \delta_{k_\ell k_{\ell+1}} \delta(k_\ell, \mu_\ell) \qquad \text{and} \qquad \boldsymbol{w}_L[k_L, \mu_L] := \delta(k_L, \mu_L).$$

Proof. Contracting the last two cores $\boldsymbol{w}_{L-1}$ and $\boldsymbol{w}_L$ leads to

$$\sum_{k_L=0}^{|\Delta_d|-1} \boldsymbol{w}_{L-1}[k_{L-1}, \mu_{L-1}, k_L] \boldsymbol{w}_L[k_L, \mu_L]$$

$$= \delta(k_{L-1}, \mu_{L-1}) \delta(k_{L-1}, \mu_L)$$

$$= \delta_{k_{L-1}0}^2 \prod_{\ell=L-1}^{L} \delta_{\mu_\ell 0} + (1 - \delta_{k_{L-1}0})^2 \prod_{\ell=L-1}^{L} (\delta_{\mu_\ell 0} + \delta_{k_{L-1}\iota_{\mu_\ell}}(1 - \delta_{\mu_\ell 0})).$$

Iterating the contraction for the remaining stochastic cores $\boldsymbol{w}_\ell$, $\ell = L-1, \dots, 1$, yields the tensor $\boldsymbol{T} \in \mathbb{R}^{|\Delta_d| \times d}$ given by

$$(3.6) \qquad \boldsymbol{T}[k_1, \mu] = \delta_{k_1 0}^L \delta_{\mu 0} + (1 - \delta_{k_1 0})^L \prod_{\ell=1}^{L} \left(\delta_{\mu_\ell 0} + \delta_{k_1 \iota_{\mu_\ell}}(1 - \delta_{\mu_\ell 0}) \right).$$

Since Δ_d is the set of univariate L-dimensional indices, for each $\mu \in \Lambda_d$ we have $\mu = \sum_{\ell=1}^{L} \mu_\ell e_\ell$ with $\mu_\ell e_\ell \in \Delta_d$. For any $\mu \in \Lambda_d \backslash \Delta_d$ there exist at least two $\ell_1 \neq \ell_2 \in \{1, \dots, L\}$ with $\mu_{\ell_1}, \mu_{\ell_2} > 0$. Since $\iota^{-1}(\mu_{\ell_1} e_{\ell_1}) \neq \iota^{-1}(\mu_{\ell_2} e_{\ell_2})$ and $0 \in \Delta_d$, i.e. $\delta_{\mu 0} = 0$, it follows that

$$\boldsymbol{T}[k_1, \mu] = \delta_{k_1 \iota_{\mu_{\ell_1}}} \delta_{k_1 \iota_{\mu_{\ell_2}}} (1 - \delta_{k_1 0})^L \prod_{\ell_1, \ell_2 \neq \ell=1}^{L} \left(\delta_{\mu_\ell 0}(1 - \delta_{k_1 \iota_{\mu_\ell}}) + \delta_{k_1 \iota_{\mu_\ell}} \right) = 0.$$

For any $\mu \in \Delta_d \backslash \{0\}$ there exists exactly one $\ell \in \{1, \dots, L\}$ such that $\mu = \mu_\ell e_\ell$ for $\mu_\ell \in [1 : d_\ell]$. Hence, $\delta_{\mu_m 0} = 1$ for all $m = \{1, \dots, L\}$ with $m \neq \ell$ and (3.6) simplifies to $\boldsymbol{T}[k_1, \mu] = (1 - \delta_{k_1 0}) \delta_{k_1 \iota^{-1}(\mu)}$. Eventually, for $\mu = 0$ we have $\iota_{\mu_\ell} = 0$ for all $\ell = 1, \dots, L$ and thus $\boldsymbol{T}[k_1, 0] = \delta_{k_1 0} \delta_{k_1 \iota^{-1}(0)}$. Combining the cases above results in $\boldsymbol{T}[k_1, \mu] = \delta_{k_1 \iota^{-1}(\mu)}$ for all $\mu \in \Delta_d$. Combining this and the definition of $\boldsymbol{w}_0$ with (3.5) concludes the proof. $\qquad \square$

Lognormal coefficient field. There exists no exact TT representation of the lognormal diffusion coefficient (2.2). Several methods to obtain an approximation of (2.2) in the TT format have been investigated [3, 6, 34, 35, 40]. Since Proposition 3.3 yields an exact representation of the affine exponent, the method of our choice is to compute the exponential of $\gamma - \gamma_0$ as proposed in [6]. In the following, we briefly recall the basic idea of this approach. The lognormal diffusion coefficient (2.2) constitutes a holonomic function, i.e. κ is the unique solution of the gradient system

$$(3.7) \qquad \begin{aligned} \nabla_y \, \kappa(x, y) &:= \kappa(x, y) \, \nabla_y (\gamma(x, y) - \gamma_0(x)), \\ \kappa(x, y_0) &:= \exp(\gamma(x, y_0) - \gamma_0(x)), \end{aligned}$$

for some arbitrary $y_0 \in \Gamma_\kappa$ and all $x \in D$. We then obtain an approximation of κ by constructing an approximation to the Galerkin projection of (3.7). For $\mathcal{V}_{N,d} := \mathcal{V}_N(\Lambda_d; \mathcal{T}, \pi_0)$, the resulting Galerkin system

$$(3.8) \qquad (A(u), v)_{L^2(\Gamma_\kappa, \pi_0; \mathbb{R}^{|\mathcal{T}|})} = (g, v)_{L^2(\Gamma_\kappa, \pi_0; \mathbb{R}^{|\mathcal{T}|})} \qquad \text{for all } v \in \mathcal{V}_{N,q+1},$$

is well defined and exhibits a unique solution $\kappa_N \in \mathcal{V}_{N,q}$, see [6, Lemma 2.1]. We then employ the *Alternating Linear Scheme* (ALS) [48] to obtain an approximation $\kappa_{N.s} \in \mathcal{V}_{N,q}$ of the Galerkin projection κ_N in the TT format. The application of the ALS is very efficient since the operator A and the right-hand side g in fact possess a TT representation with ranks bounded independently of the problem dimension L. Additionally, this approach allows for an a posteriori approximation error estimation with respect to the induced energy norm $\|\bullet\|_A := \|A(\bullet)\|_{L^2(\Gamma_\kappa, \pi_0; \mathbb{R}^{|\mathcal{T}|})}$.

Theorem 3.4 ([6, Theorem 2.4]). *For $L/2 + 1 \leqslant k \in \mathbb{N}$ let $\kappa \in H^k(\Gamma_\kappa, \pi_0; L^\infty(D))$ be the unique solution of (3.7) and let $w_N \in \mathcal{V}_{N,q}$ be arbitrary. Then it holds*

$$\|\mathcal{R}(w_N)\|_{\mathcal{V}_{N,q+1}^*} \leqslant \|\kappa - w_N\|_A \leqslant \|\mathcal{R}(w_N)\|_{\mathcal{V}_{N,q+1}^*} + \|\Pi_{\mathcal{V}_{N,q+1}^\perp} g\|_{L^2(\Gamma_\kappa, \pi_0; L^\infty(D))},$$

where $\mathcal{R}(w_N) = g - A(w_N) \in \mathcal{V}_{N,q+1}^*$ denotes the residual and $\Pi_{\mathcal{V}_{N,q+1}^\perp}$ is the projection onto the orthogonal complement of $\mathcal{V}_{N,q+1}$.

Note that it is possible to construct the gradient system (3.7) in a way such that $g \in \mathcal{V}_{N,q+1}$, which directly implies $\|\Pi_{\mathcal{V}_{N,q+1}^\perp} g\|_{L^2(\Gamma_\kappa, \pi_0; L^\infty(D))} = 0$. Moreover, we know from [6, Lemma 2.3] that $\|\boldsymbol{g} - \boldsymbol{A}\boldsymbol{\kappa}\|_2 = \|\mathcal{R}(\kappa_{N,s})\|_{\mathcal{V}_{N,q+1}^*}$, where $\|\bullet\|_2$ denotes the Frobenius norm and $\boldsymbol{A}, \boldsymbol{g}$ and $\boldsymbol{\kappa}$ are the coefficient tensors of A, g and $\kappa_{N,s}$, respectively. With this and since Theorem 3.4 holds for any discrete $w_N \in \mathcal{V}_N(\Lambda_q; \mathcal{T}, \pi_0)$, this implies that the computed residual of the ALS effectively bounds the energy approximation error from above and below. This can be used in the adaptive algorithm to ensure that the approximation $\kappa_{N,s}$ of κ is sufficiently accurate.

Remark 3.5. *We utilize the simple structure of γ, κ and f to generate efficient low-rank approximations. However, it should be pointed out that there is no restriction per se and the used approximation technique can readily be applied to more complicated problems or alternative approximation techniques can be used as a substitute without further adaptation of other parts of our approach. As an example, one could consider non-intrusive reconstruction techniques such as a TT cross approximation [41] or a VMC reconstruction if the diffusion field is only accessible by pointwise evaluations.*

4. ERROR ESTIMATION

In this section we recall the residual based error estimator presented in [3], which is an adaptation of the development in [1, 8]. Additionally, we motivate an heuristic indicator to steer the number of regression samples in the adaptive algorithm. The results are stated with the lognormal case (B) in mind, but equally hold true without any adaptation for the simpler affine coefficient, using the notation described in case (A).

4.1. **Residual based error estimator.** We note that there exists no exact TT representation of the lognormal diffusion coefficient, thus the approximation of κ described in Section 3.2 introduces an additional error. Since this error can be controlled independently, we assume the approximation of κ to be sufficiently accurate such that it can be neglected henceforth.

In the following we assume some fixed FE polynomial degree $p \in \mathbb{N}$ and consider $M \leqslant L \in \mathbb{N}$. Furthermore, let $d, q \in \mathcal{F}$ with $\text{supp}(d) = \{1, \ldots, M\} \subseteq \{1, \ldots, L\} = \text{supp}(q)$. Assume $w_N \in$

$\mathcal{V}_N = \mathcal{V}_N(\Lambda_d; \mathcal{T}, p)$ is given by the TT representation

$$w_N(x,y) := \sum_{j \in [N]} \sum_{\mu \in \Lambda_d} \Big(\sum_{k=1}^{r} \boldsymbol{w}_0[j, k_1] \prod_{m=1}^{M} \boldsymbol{w}_m[k_m, \mu_m, k_{m+1}] \Big) \varphi_j(x) P_\mu(y)$$

with ranks $r \in \mathbb{N}^M$. Similarly, assume that the coefficient field $a_N \in L^2(\Gamma, \pi; L^\infty(D))$ is given in a semi-discretized form

$$a_N(x,y) := \sum_{\nu \in \Lambda_q} \Big(\sum_{k=1}^{s} a_0[k_1](x) \prod_{\ell=1}^{L} \boldsymbol{a}_\ell[k_\ell, \nu_\ell, k_{\ell+1}] \Big) P_\nu(y)$$

with ranks $s \in \mathbb{N}^L$ and $a_0[k_1] \in \mathcal{X}$ for all $k_1 = 1, \ldots, s_1$. Define the residual $\mathcal{R}(v) \in \mathcal{V}^* = L^2(\Gamma, \pi; \mathcal{X}^*)$ of (2.5) by $\mathcal{R}(v) := F - B(v, \bullet)$. The energy error can then be bounded in the following way.

Theorem 4.1 ([1, Theorem 5.1]). *Let $\mathcal{V}_N \subset \mathcal{V}$ be a closed subspace and $w_N \in \mathcal{V}_N$ arbitrary. Let u_N denote the Galerkin projection with respect to B of u onto $\mathcal{V}_N$. It then holds that*

$$\|u - w_N\|_B^2 \leqslant \left(\sup_{v \in \mathcal{V} \backslash \{0\}} \frac{|\langle \mathcal{R}(w_N), (1 - I_{\mathrm{C}})v \rangle_{\mathcal{V}^*, \mathcal{V}}|}{\check{c} \, \| \nabla v \|_{\pi_0, D}} + c_I \|u_N - w_N\|_B \right)^2 + \|u_N - w_N\|_B^2.$$

Here, I_{C} denotes the tensor product interpolation operator defined in [3], c_I is the operator norm of $1 - I_{\mathrm{C}}$ with respect to $\| \bullet \|_B$ and $\check{c}$ is the coercivity constant of the bilinear form B.

We point out, that showing coercivity of the bilinear form for the lognormal case (B) is not trivial, see Appendix B for more details and references. Since we consider $w_N \in \mathcal{V}_N$ as well as a finite expansion a_N of the diffusion coefficient a, the residual $\mathcal{R}(w_N)$ is itself characterized by a finite polynomial expansion. In particular, we have $\mathcal{R}(w_N) = f + \mathrm{div}(r(w_N))$ for $r(w_N) := a_N \nabla w_N$. The semi-discrete expansion

$$(4.1) \qquad r(w_N)(x,y) = \sum_{\mu \in \Lambda_{d+q-1}} r_\mu(w_N)(x) P_\mu(y),$$

with $r_\mu(w_N)$ derived from the TT representations of a_N and w_N, allows to split the residual into an active and an inactive part. With the inactive set $\Delta := \Lambda_{(d+q-1)} \backslash \Lambda_d$, consider the splitting $\mathcal{R}(w_N) = \mathcal{R}_{\Lambda_d}(w_N) + \mathcal{R}_\Delta(w_N)$ for

$$\mathcal{R}_{\Lambda_d}(w_N) := f + \sum_{\mu \in \Lambda_d} \mathrm{div}(r_\mu(w_N)) P_\mu \qquad \text{and} \qquad \mathcal{R}_\Delta(w_N) := \sum_{\mu \in \Delta} \mathrm{div}(r_\mu(w_N)) P_\mu.$$

Deterministic estimator contributions. The active part of the residual is associated with the deterministic approximation error of the FE discretization. This enables to estimate the error on each triangle for the active set. Define for any $w_N \in \mathcal{V}_N$ the deterministic error estimator contribution

$$(4.2) \qquad \eta_{\det}(w_N, \mathcal{T}, \Lambda_d)^2 := \sum_{T \in \mathcal{T}} \eta_{\det, T}(w_N, \Lambda_d)^2 + \sum_{E \in \mathcal{E}} \eta_{\det, E}(w_N, \Lambda_d)^2,$$

where the volume and edge terms respectively read

$$(4.3) \qquad \eta_{\det, T}(w_N, \Lambda_d) := h_T \|\mathcal{R}_{\Lambda_d}(w_N) \zeta_{\vartheta\rho}\|_{\pi_0, T} \qquad \qquad \text{for all } T \in \mathcal{T},$$

$$(4.4) \qquad \eta_{\det, E}(w_N, \Lambda_d) := h_E^{1/2} \| \sum_{\mu \in \Lambda_d} [\![r_\mu]\!]_E P_\mu \, \zeta_{\vartheta\rho}\|_{\pi_0, E} \qquad \text{for all } E \in \mathcal{E}.$$

The deterministic estimator contribution bounds the active part of the residual as the following lemma shows.

Lemma 4.2 ([3, Proposition 5.3]). *For any $v \in \mathcal{V}$ and any $w_N \in \mathcal{V}_N$, it holds that*

$$|\langle \mathcal{R}_{\Lambda_d}(w_N), (1 - I_{\mathrm{C}})v \rangle_{\mathcal{V}*,\mathcal{V}}| \leqslant c_{\det}\eta_{\det}(w_N, \mathcal{T}, \Lambda_d) \| \nabla v \|_{\pi_0,D}.$$

Stochastic estimator contributions. The inactive part of the residual is associated with the approximation error due to the truncation of the polynomial expansion. This can be used to obtain a bound for the residual on the inactive part. Define for any $w_N \in \mathcal{V}_N$ the stochastic error estimator

$$(4.5) \qquad \eta_{\mathrm{sto}}(w_N, \Delta) := \| \sum_{\mu \in \Delta} r_\mu(w_N) P_\mu \zeta_{\vartheta\rho} \|_{\pi_0,D}.$$

Again, the stochastic error is bounded for the affine as well as the lognormal case.

Lemma 4.3 ([3, Proposition 5.5]). *For any $v \in \mathcal{V}$ and any $w_N \in \mathcal{V}_N$, it holds that*

$$|\langle \mathcal{R}_{\Delta}(w_N), (1 - I_{\mathrm{C}})v \rangle_{\mathcal{V}*,\mathcal{V}}| \leqslant \eta_{\mathrm{sto}}(w_N, \Delta) \, \| \nabla v \|_{\pi_0,D}.$$

To obtain a localization that can be used in the adaptive refinement strategy, we split η_{sto} into different parts each providing information about the influence of the individual stochastic modes. For all $\ell = 1, \dots, L$ and *look-ahead* $t_\ell \in [1 : q_\ell - 1]$, consider the index sets of uncoupled parameters

$$\Delta_{\ell,t_\ell} := \bigotimes_{m=1}^{\ell-1} [d_m] \otimes [d_\ell : d_\ell + t_\ell] \otimes \bigotimes_{m=\ell+1}^{L} [d_m] \otimes [1]\dots$$

The look ahead t_ℓ allows for more information about the behaviour of the respective stochastic dimension than the index sets Ξ_ℓ considered in [3]. The local stochastic estimator contributions $\eta_{\mathrm{sto}}^2(w_N, \Delta_{\ell,t_\ell})$ are then defined as in (4.5) for the sets Δ_{ℓ,t_ℓ}.

Algebraic estimator contributions. The algebraic error $\|u_N - w_N\|_B$ incorporates the distance of w_N to the $\mathcal{V}_N$ best approximation u_N. Since we employ the sample based VMC regression to obtain an approximation $u_{N,r,n}$ of u_N, this error can be used as an indicator to control the number of VMC samples to guarantee that $u_{N,r,n}$ is a quasi-best approximation with high probability. The algebraic error can be bounded by the quantity

$$(4.6) \qquad \eta_{\mathrm{alg}}(w_N) := \|(\boldsymbol{B}\boldsymbol{w} - \boldsymbol{f})\boldsymbol{S}^{-1/2}\|_2,$$

where $\| \bullet \|_2$ denotes the Frobenius norm, $\boldsymbol{f}$ is the coefficient tensor of the right-hand side f in $\mathcal{V}_N$ and $\boldsymbol{B}$ is the discrete version of the operator induced by (2.4). The rank-one base change tensor $\boldsymbol{S}$ translates integrals of Hermite polynomials with respect to the measure π to π_0 and is given by the components $\boldsymbol{S}_0[i, j] := \int_D \nabla \varphi_i \cdot \nabla \varphi_j \, \mathrm{d}x$ for the spatial dimension and

$$(4.7) \qquad \boldsymbol{S}_m[\mu_m, \mu'_m] := \int_{\Gamma_m} P_{\mu_m}^m (y_m) P_{\mu'_m}^m (y_m) \, \mathrm{d}\pi_m(y_m; \sigma(0))$$

for the stochastic modes. In the affine case (A) this implies $\boldsymbol{S}_m = I_{d_m}$. The algebraic error is bounded in the following way.

Lemma 4.4 ([3, Proposition 5.6]). *For any $w_N \in \mathcal{V}_N$ and the Galerkin solution $u_N \in \mathcal{V}_N$ of (2.7), it holds that*

$$\|u_N - w_N\|_B \leqslant \check{c}^{-1}\eta_{\mathrm{alg}}(w_N).$$

Combined error estimator. As a corollary of Theorem 4.1 and Lemmas 4.2–4.4, the energy error can be bounded by the combined overall error estimator

$$(4.8) \qquad \eta(w_N)^2 := \big(c_{\det}\eta_{\det}(w_N, \mathcal{T}, \Lambda_d) + \eta_{\mathrm{sto}}(w_N, \Delta) + c_I\eta_{\mathrm{alg}}(w_N)\big)^2 + \eta_{\mathrm{alg}}(w_N)^2.$$

Corollary 4.5 ([3, Corollary 5.7]). *For any $w_N \in \mathcal{V}_N$ it holds $\|u - w_N\|_B \leqslant \check{c}^{-1} \eta(w_N)$.*

4.2. **Regression error indicator.** Since the proposed method fundamentally relies on solving the nonlinear least squares problem (3.3) to compute a tensor representation of the solution, it is important to adapt the number of samples in order to guarantee a robust approximation. In this section we discuss a heuristic indicator for doing this when (3.3) is solved by means of an ALS method. ALS methods are part of a family of iterative methods that minimize (3.3) by solving a sequence of least squares problems

$$(4.9) \qquad w_{N,k,n} := \underset{w_N \in \mathcal{W}_k}{\arg\min} \|u_N - w_N\|_n$$

on linear subsets $\mathcal{W}_k \subseteq \mathcal{M}_r$ for $k \in \mathbb{N}$. Note that $w_{N,k,n}$ is an empirical estimate of the best approximation $w_{N,k}$ of u_N in the linear space $\mathcal{W}_k$. The ensuing estimation error can be bounded by the subsequent lemma that is adapted from [51].

Lemma 4.6. *Let $\{P_1, \ldots, P_D\}$ be any orthonormal basis of $\mathcal{W}_k$ and let $G_n \in \mathbb{R}^{D \times D}$ be the empirical Gramian given by $(G_n)_{ij} := \frac{1}{n} \sum_{t=1}^{n} P_i(y^{(t)}) P_j(y^{(t)})$. If the smallest eigenvalue of G_n satisfies $\lambda_{\min}(G_n) > 0$, then*

$$\|u_N - w_{N,k,n}\|_{L^2(\Gamma,\pi_\rho;\mathcal{X})} \leqslant (1 + \lambda_{\min}(G_n)^{-1/2}) \|u_N - w_{N,k}\|_{L^\infty(\Gamma,\pi_\rho;\mathcal{X})}.$$

Proof. In the following we abbreviate $\| \bullet \|_{\infty,D} := \| \bullet \|_{L^\infty(\Gamma,\pi_\rho;L^2(D))}$. Let $v \in \mathcal{W}_k$ be arbitrary and let $\boldsymbol{v}$ denote the coefficients of v with respect to the basis $\{P_1, \ldots, P_D\}$. Now note that $\|v\|_n^2 = \boldsymbol{v}^{\mathsf{T}} G_n \boldsymbol{v} \geqslant \lambda_{\min}(G_n) \| \nabla_x v\|_{\pi_\rho,D}^2$ is a Hilbert space norm and that $w_{N,k,n}$ is the orthogonal projection of u_N onto the space $\mathcal{W}_k$ with respect to the corresponding inner product. Hence,

$$(4.10) \qquad \|u - v\|_n^2 = \|u - w_{N,k,n}\|_n^2 + \|w_{N,k,n} - v\|_n^2 \geqslant \|w_{N,k,n} - v\|_n^2.$$

Combining this observation with the lower bound $\| \nabla_x v\|_{\pi_\rho,D} \leqslant \lambda_{\min}(G_n)^{-1/2} \|v\|_n$ yields

$$\| \nabla_x (u - w_{N,k,n})\|_{\pi_\rho,D} \leqslant \| \nabla_x(u - v)\|_{\pi_\rho,D} + \lambda_{\min}(G_n)^{-1/2} \|v - w_{N,k,n}\|_n$$
$$\leqslant \| \nabla_x(u - v)\|_{\pi_\rho,D} + \lambda_{\min}(G_n)^{-1/2} \|u - v\|_n$$
$$\leqslant (1 + \lambda_{\min}(G_n)^{-1/2}) \| \nabla_x(u - v)\|_{\infty,D}.$$

The first inequality is simply the triangle inequality and the final inequality follows from the fact that both $\| \nabla_x \bullet \|_{\pi_\rho,D}$ and $\| \bullet \|_n$ are dominated by $\| \nabla_x \bullet \|_{\infty,D}$. Since v is arbitrary, we can substitute $v = w_{N,k}$ to prove Lemma 4.6. $\qquad\square$

Assuming $\lambda_{\min}(G_n) \geqslant \varepsilon > 0$, Lemma 4.6 states that the error of the least squares approximation can be bounded up to the constant factor of $(1 + \varepsilon^{-1/2})$ by the best approximation error. The condition $\lambda_{\min}(G_n) \geqslant \varepsilon$ thus provides an indicator for the necessity of increasing the number of samples, which is summarized in the following algorithm.

Algorithm 1: Increase number of VMC samples (`update_samples`)

Input: initial samples $\{y^{(i)}\}_{i=1}^n$; stochastic dimensions d; ratio $\theta_{\mathrm{alg}} > 0$; minimum eigenvalue threshold ε_G

1   assemble polynomial tensor basis $\{P_1, \ldots, P_D\}$ from dimension tuple d;

2   compute G_n as in Lemma (4.6);

3   **while** $\lambda_{\min}(G_n) < \varepsilon_G$ **do**

4   draw $\lfloor \theta_{\mathrm{alg}} n \rfloor$ new samples $\{y_{\mathrm{new}}^{(j)}\}$;

5      $\{y^{(i)}\} \leftarrow \{y^{(i)}\} \cup \{y_{\mathrm{new}}^{(j)}\}$;

6   update G_n;

7   **return** *samples* $\{y^{(i)}\}$

Note that Lemma 4.6 is only valid for the linear sub-problems (4.9) and that the bound is not applicable to the global nonlinear problem (3.3), but any attempt to integrate this criterion requires extensive intervention with exiting code. To comply with our objective of non-intrusiveness, we regard the optimization algorithm as a black box and apply Algorithm 1 only to the final result. We note that this result may be unrelated to the local solutions $w_{N,k,n}$. Hence, this heuristic cannot guarantee that sufficiently many samples are available in each step of the optimization. However, in our experiments this never seemed to cause any problems.

Similar to [3], we rely on η_{alg} to control the error caused by the approximation of the Galerkin solution due to a black-box solver, in particular because of an iterative algebraic solver and the low-rank compression. Due to the rank-adaptive nature of the employed recovery method, we only need to consider a strategy to increase the number of regression samples if η_{alg} dominates. Nevertheless, note that $\eta_{\mathrm{alg}} = \|(\boldsymbol{B}\boldsymbol{u} - \boldsymbol{f})\boldsymbol{S}^{-1/2}\|_2$ is a discretized form of the residual and thus contains more than just the error $\|u_N - u_{N,r,n}\|_B$ of the regression. We hence apply the heuristic indicator to check if more samples are required or if the stochastic space needs to be enlarged.

Algorithm 2: Mark & Refine (`mark_and_refine`)

Input: mesh $\mathcal{T}$; stochastic dimensions d; samples $\{y^{(i)}\}$; global estimator contributions
$\qquad \eta_{\mathrm{det}}(w_N, \mathcal{T}, \Lambda_d), \eta_{\mathrm{sto}}(w_N, \Delta), \eta_{\mathrm{alg}}(w_N)$; local estimator contributions $\eta_{\mathrm{det},T}(w_N, \Lambda_d)$,
$\qquad \eta_{\mathrm{det},E}(w_N, \Lambda_d), \eta_{\mathrm{sto}}(w_N, \Delta_{m,t_m})$; Dörfler thresholds $0 < \theta_{\mathrm{det}}, \theta_{\mathrm{sto}} \leqslant 1$; ratio $\theta_{\mathrm{alg}} > 0$;
$\qquad$ minimal eigenvalue threshold ε_G

1 **if** $\eta_{\mathrm{det}}(w_N, \mathcal{T}, \Lambda_d) = \max\{\eta_{\mathrm{det}}(w_N, \mathcal{T}, \Lambda_d), \eta_{\mathrm{sto}}(w_N, \Delta), \eta_{\mathrm{alg}}(w_N)\}$ **then**

2 $\quad$ choose minimal set $\mathcal{M} \subset \mathcal{T}$, such that $\eta_{\mathrm{det}}(w_N, \mathcal{M}, \Lambda_d) \geqslant \theta_{\mathrm{det}}\eta_{\mathrm{det}}(w_N, \mathcal{T}, \Lambda_d)$;

3 $\quad \mathcal{T} \leftarrow \texttt{bisect}(\mathcal{T}, \mathcal{M})$;

4 **else if** $\eta_{\mathrm{sto}}(w_N, \Delta) = \max\{\eta_{\mathrm{det}}(w_N, \mathcal{T}, \Lambda_d), \eta_{\mathrm{sto}}(w_N, \Delta), \eta_{\mathrm{alg}}(w_N)\}$ **then**

5 $\quad$ choose minimal set $\mathcal{M} \subset \mathbb{N}$, such that $\sum_{m \in \mathcal{M}} \eta_{\mathrm{sto}}(w_N; \Delta_{m,t_m}) \geqslant \theta_{\mathrm{sto}}\eta_{\mathrm{sto}}(w_N, \Delta)$;

6 $\quad d_m \leftarrow d_m + 1$ for $m \in \mathcal{M}$;

7 **else**

8 $\quad$ compute G_n as in Lemma (4.6);

9 $\quad$ **if** $\lambda_{\min}(G_n) < \varepsilon_G$ **then**

10 $\quad\quad \{y^{(i)}\} \leftarrow \texttt{update\_samples}(\{y^{(i)}\}, d, \theta_{\mathrm{alg}}, \varepsilon_G)$

11 $\quad$ **else**

12 $\quad\quad$ choose minimal set $\mathcal{M} \subset \mathbb{N}$, such that $\sum_{m \in \mathcal{M}} \eta_{\mathrm{sto}}(w_N; \Delta_{m,t_m}) \geqslant \theta_{\mathrm{sto}}\eta_{\mathrm{sto}}(w_N, \Delta)$;

13 $\quad\quad d_m \leftarrow d_m + 1$ for $m \in \mathcal{M}$;

14 **return** $\mathcal{T}, d, \{y^{(i)}\}$

---

5. ALGORITHMIC REALIZATION

This section describes the algorithm that steers the adaptive refinement of the FE space, the stochastic expansion size and the VMC sampling and compression errors. Given the approximation $u_{N,r,n} \in \mathcal{V}_N$ of the solution obtained via the VMC approach and the diffusion coefficient $a_N \in \mathcal{V}_N(\Lambda_q; \mathcal{T}, p)$ in TT format, the estimator contributions can be computed efficiently as detailed in [3].

With the combined error estimator from Corollary 4.5 it is possible to define an algorithm that automatically steers the refinement of the spatial triangulation $\mathcal{T}$, adds stochastic modes or increases mode dimensions if necessary and ensures that the low-rank representation error of the solution $u_{N,r,n}$ is not predominant. The adaptive algorithm described in this section is based on the one presented in [1,

3, 8] and consists in principle of three major steps, namely *SOLVE*, *ESTIMATE* and *MARK & REFINE*, which are iterated until some stopping condition is satisfied.

Given some triangulation $\mathcal{T}$, a fixed FE polynomial degree $p \in \mathbb{N}_0$ and $d, q \in \mathcal{F}$ with $\mathrm{supp}(d) \subset \mathrm{supp}(q)$, the *SOLVE* step generates low-rank approximations of the solution $u_{N,r,n} \in \mathcal{V}_N(\Lambda_d; \mathcal{T}, p)$, the diffusion coefficient $a_N \in \mathcal{V}_N(\Lambda_q; \mathcal{T}, p)$ and the right-hand side $f \in \mathcal{V}_N(\Lambda_d; \mathcal{T}, p = 0)$. The solution u_N is reconstructed from N_{VMC} many training samples $(y^{(i)}, u^{(i)})$ using the VMC method. To obtain an approximation of the diffusion coefficient, we use Proposition 3.3 to construct the exact representation of the affine field (2.1). Then, depending on the coefficient type, we either use the exact TT representation directly or compute the Galerkin projection of (3.7) via the $\mathtt{ExpTT}$ algorithm presented in [6, Algorithm 2]. When employing the $\mathtt{ExpTT}$ algorithm, we ensure that the error in Theorem 3.4 is sufficiently small by automatically decreasing the rounding threshold and increasing the projection dimensions during rescaling until a certain tolerance is reached. Since we assume a right-hand side independent of y, Proposition 3.3 also yields an exact representation of f in the TT format. We note that the TT representations $u_{N,r,n}$, a_N and f in case (B) are not with respect to the correctly scaled Hermite basis. Thus a subsequent transformation to the correct basis has to be performed.

Algorithm 3: Adaptive non-intrusive algorithm (aVMC)

Input: initial mesh $\mathcal{T}_0$; FE polynomial degree p; initial stochastic dimensions d; initial samples $\{y^{(i)}\}$; Dörfler thresholds $0 < \theta_{\mathrm{det}}, \theta_{\mathrm{sto}} \leqslant 1$; ratio $\theta_{alg} > 0$; max. number of iterations N_{iter}; max. number of TT-DoFs N_{TT}; accuracy ε; minimal eigenvalue threshold ε_G;

Output: solution $u_{N,r,n}$; combined estimator $\eta(u_{N,r,n})$ for each iteration;

1 **for** $j = 1, \ldots, N_{\mathrm{iter}}$ **do**
2   *SOLVE*
3   generate training samples $\{u^{(i)} = u(x, y^{(i)})\}$ for $i = 1, \ldots, N_{\mathrm{VMC}}$;
4          $u_{N,r,n} \leftarrow \mathtt{VMC}\big(\mathcal{V}_N(\Lambda_d; \mathcal{T}, p), \{y^{(i)}\}, \{u^{(i)}\}\big)$;
5   construct γ and f according to Proposition 3.3;
6          $a_N \leftarrow \gamma$ in case (A)   **OR** $a_N \leftarrow \mathtt{ExpTT}(\gamma, q, y_0 = 0, \mathrm{tol} = 10^{-5})$ in case (B);
7   *ESTIMATE*
8   compute $\eta_{\mathrm{det}}(u_{N,r,n}, \mathcal{T}, \Lambda_d)$, $\big(\eta_{\mathrm{det}}(u_{N,r,n}, \{T\}, \Lambda_d)\big)_{T \in \mathcal{T}}$ according to (4.2)–(4.4);
9   compute $\eta_{\mathrm{sto}}(u_{N,r,n}, \Delta)$, $\big(\eta_{\mathrm{sto}}(u_{N,r,n}; \Delta_{m,t_m})\big)_{m \in \mathbb{N}}$ according to (4.5);
10   compute $\eta_{\mathrm{alg}}(u_{N,r,n})$ according to (4.6);
11   compute $\eta_j = \eta_j(u_{N,r,n})$ according to Corollary 4.5;
12   **if** $\eta_j(u_{N,r,n}) < \varepsilon$ *or* $\mathrm{tt}\text{-}\mathrm{dofs}(u_{N,r,n}) > N_{\mathrm{TT}}$ **then**
13   **break**
14   *MARK & REFINE*
15          $\eta_{\mathrm{glob}} := (\eta_{\mathrm{det}}(u_{N,r,n}, \mathcal{T}, \Lambda_d), \eta_{\mathrm{sto}}(u_{N,r,n}, \Delta), \eta_{\mathrm{alg}}(u_{N,r,n}))$;
16          $\eta_{\mathrm{loc}} := \Big(\big(\eta_{\mathrm{det}}(u_{N,r,n}, \{T\}, \Lambda_d)\big)_{T \in \mathcal{T}}, \big(\eta_{\mathrm{sto}}(u_{N,r,n}; \Delta_{m,t_m})\big)_{m \in \mathbb{N}}\Big)$;
17          $\mathcal{T}, d, \{y^{(i)}\} \leftarrow \mathtt{mark\_and\_refine}(\mathcal{T}, d, \{y^{(i)}\}, \eta_{\mathrm{glob}}, \eta_{\mathrm{loc}}, \theta_{\mathrm{det}}, \theta_{\mathrm{sto}}, \theta_{\mathrm{alg}}, \varepsilon_G)$
18 **return** $u_{N,r,n}, (\eta_1, \eta_2, \ldots)$

The local and global estimator contributions according to Section 4 are computed in the *ESTIMATE* step. We emphasize that the global contributions $\eta_{\mathrm{det}}(u_{N,r,n}, \mathcal{T}, \Lambda_d)$ and $\eta_{\mathrm{sto}}(u_{N,r,n}, \Delta)$ are used to determine the refinement strategy, but that we use localized versions $\eta_{\mathrm{det}}(u_{N,r,n}, \{T\}, \Lambda_d)$, $T \in \mathcal{T}$, and $\eta_{\mathrm{sto}}(u_{N,r,n}, \Delta_{\ell,t_\ell})$, $\ell = 1, \ldots, M+1$, to determine which triangles and modes are refined.

The *MARK & REFINE* step uses the estimator contributions in combination with different selection criteria to enlarge the FE space, the stochastic dimensions or the number of training samples for the VMC algorithm, respectively. We employ a Dörfler marking strategy for the spatial and stochastic refinement and a simple relative increase in the number of VMC samples as described in Algorithm 2. For the refinement of the spatial mesh $\mathcal{T}$ we use newest vertex bisection [52] on all marked elements $T \in \mathcal{M}$, which is denoted by $\texttt{bisect}(\mathcal{T}, \mathcal{M})$ in Algorithm 2.

Algorithm 3 iterates the solve, estimate, mark and refine loop until the combined estimator $\eta(u_N)$ is sufficiently small or a maximum problem size is reached. Note that the algorithmic realization of the VMC reconstruction is abbreviated by the $\texttt{VMC}$ method in line 4.

6. NUMERICAL EXPERIMENTS

In this section we examine the performance of the adaptive algorithm for benchmark problems similar to [3, 49]. Tensor calculus is carried out with the open source software package $\texttt{xerus}$ [53]. Finite element computations to generate training samples are conducted with the $\texttt{FEniCS}$ package [54]. As spatial domain we choose either the unit square $D = [0,1]^2$ or the L-shaped domain $D = [0,1]^2 \backslash (0.5, 1)^2$. The derived total error estimator η is used to steer the adaptive refinement of the triangulation $\mathcal{T}$, the space Λ_d and the number of reconstruction samples N_{VMC}. We investigate the behaviour of the individual estimator contributions and how they influence the true (sampled) expected energy error. Moreover, we comment on the complexity of the coefficient discretization.

6.1. **Computation of the error.** To validate the reliability of the estimator and its contributions in the adaptive scheme, we compute an empirical approximation of the true $L^2(\Gamma, \pi; \mathcal{X})$-error using N_{MC} samples, i.e.

$$\mathcal{E}_u(u_{N,r,n})^2 := \frac{1}{N_{\mathrm{MC}}} \sum_{i=1}^{N_{\mathrm{MC}}} \| u(y^{(i)}) - u_{N,r,n}(y^{(i)}) \|_{\mathcal{X}(\hat{\mathcal{T}})}^2 \approx \| \nabla(u - u_{N,r,n}) \|_{\pi,D}^2.$$

Here, the parametric solution $u_{N,r,n} \in \mathcal{V}_N(\Lambda_d; \mathcal{T}, p)$ is compared to the deterministic sampled solution $u(y^{(i)})$ projected onto a uniform refinement $\hat{\mathcal{T}}$ of the finest FE mesh obtained in the adaptive refinement loop. Since all triangulations generated by Algorithm 3 as well as $\hat{\mathcal{T}}$ are nested, we employ simple nodal interpolation of each $u_{N,r,n}$ onto $\hat{\mathcal{T}}$ to guarantee $u_{N,r,n} \in \mathcal{V}_N(\Lambda_d; \hat{\mathcal{T}}, p)$. Note that the reference samples $u(y^{(i)})$ are computed by the parametric black-box solver of the forward problem, i.e., no functional approximation of u on $\hat{\mathcal{T}}$ is required to compute $\mathcal{E}_u(u_{N,r,n})$. All estimator contributions depend on the diffusion coefficient and the right-hand side. To guarantee that the low-rank approximation errors of a and f have a negligible impact, we monitor the relative empirical L^2-L^∞ error given by

$$\varepsilon_w^\infty(w_N)^2 := \frac{\sum_{i=1}^{N_{\mathrm{MC}}} \| w(y^{(i)}) - w_N(y^{(i)}) \|_{L^\infty(D)}^2}{\sum_{i=1}^{N_{\mathrm{MC}}} \| w(y^{(i)})) \|_{L^\infty(D)}^2}, \qquad \text{for all } w, w_N \in L^2(\Gamma, \pi; L^\infty(D)).$$

The choice of $N_{\mathrm{MC}} = 250$ proved to be sufficient to obtain consistent estimates of the error in our experiments.

6.2. **The stochastic model problem.** In the numerical experiments, we consider the stationary diffusion problem (1.1) with constant right-hand side $f(x,y) = 1$. For the affine diffusion field, the expansion coefficients γ_ℓ enumerate planar Fourier modes in increasing total order and are given by $\gamma_0(x) = 1$ and

$$\gamma_\ell(x) = \frac{9}{10\zeta(\sigma)} \ell^{-\sigma} \cos\big(2\pi\beta_1(\ell)x_1\big) \cos\big(2\pi\beta_2(\ell)x_2\big), \qquad \ell = 1, \ldots, L,$$

where ζ is the Riemann zeta function and, for $k(\ell) = \lfloor -\frac{1}{2} + \sqrt{\frac{1}{4} + 2\ell} \rfloor$, $\beta_1(\ell) = \ell - k(\ell)(k(\ell)+1)/2$ and $\beta_2(\ell) = k(\ell) - \beta_1(\ell)$. For our experiments we consider a slow ($\sigma = 2$) and a fast ($\sigma = 4$) decay rate to test if the adaptive algorithm captures the relevance of the higher order modes. The spatial discretization of the diffusion coefficient is the same space as that of the solution, i.e. conforming Lagrange elements of order $p = 1$ or $p = 3$. For the lognormal case (B), we choose $\rho = 1$ and $\vartheta = 0.1$ similar to [3].

6.3. **Tensor train representation of the diffusion coefficient.** The affine diffusion coefficient γ and the constant right-hand side f can be represented in TT format as described in Proposition 3.3.

An approximation $\kappa_{N,s}$ of the lognormal coefficient field is computed via the Galerkin projection described in Section 3.2. In particular, we employ a scaling trick to account for numerical instabilities caused by the global polynomial approximation of an exponential function. We choose a scaling $t \in \mathbb{N}$ and construct an approximation $\kappa_{N,s,t} \approx \exp(2^{-t}(\gamma - \gamma_0))$ via the Galerkin approach. The diffusion coefficient $\kappa_{N,s}$ can then be recovered by squaring $\kappa_{N,s,t}$ t times. For a detailed description of the process we refer to the algorithms in [6].

As scaling constant we choose $t = 4$ and round $\kappa_{N,s,t}$ to a precision of 10^{-6} and project the stochastic components onto Hermite polynomials of uniform maximum degree 10 in each squaring step to reduce storage requirements. The ALS algorithm computing the approximation $\kappa_{N,s,t}$ has a termination threshold of 10^{-8} that needs to be reached in the Frobenius norm of the difference of two successive iteration results. This results in a relative approximation error $\varepsilon_\kappa^\infty(\kappa_{N,s}) < 10^{-4}$, which is at least one order of magnitude smaller then that of the solution $u_{N,r,n}$ in all experiments.

We also note that the exact TT representation of γ has uniform ranks $s = L + 1$, which results in large storage requirements throughout the computations. Such large representation ranks are, however, not necessary for a sufficient numerical treatment, since the ranks decrease drastically upon rounding γ to machine precision. Reducing the ranks of γ crucially decreases the required memory of both the estimator contributions and the operator A in (3.8), which is a limiting factor otherwise. A detailed investigation of the approximation quality and representation ranks of $\kappa_{N,s}$ can be found in [6].

6.4. **Adaptive convergence results.** The fully adaptive Algorithm 3 is instantiated with a single stochastic dimension $M = 1$ discretized with a linear polynomial, i.e. $d_1 = 2$. The initial spatial mesh consists of $|\mathcal{T}_0| = 143$ triangles with $p = 1$ ansatz functions and $|\mathcal{T}_0| = 64$ with $p = 3$ for the L-shaped domain. For the initial triangulations of the unit square domain, we choose $|\mathcal{T}_0| = 205$ triangles with $p = 1$ ansatz functions and $|\mathcal{T}_0| = 76$ with $p = 3$. The marking parameters are set to $\theta_{\mathrm{det}} = 0.3$, $\theta_{\mathrm{sto}} = 0.5$ and $\theta_{\mathrm{alg}} = 0.3$, respectively.
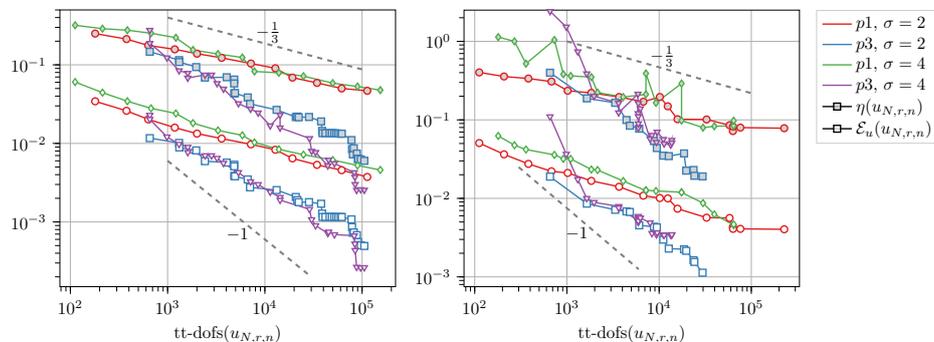

FIGURE 1. Sampled root mean square error $\mathcal{E}_u(u_{N,r,n})$ and total error estimator $\eta(u_{N,r,n})$ of the fully adaptive algorithm on the L-shaped domain. Considered are finite element approximations of order $p = 1$ and $p = 3$ for slow ($\sigma = 2$) and fast ($\sigma = 4$) decay for the affine (left) and lognormal (right) case.

Figure 1 depicts the true sampled root mean squared $H_0^1(D)$ error $\mathcal{E}_u(u_{N,r,n})$ and the corresponding overall error estimator $\eta(u_{N,r,n})$ for the affine and lognormal case, respectively. Depicted are combinations of slow ($\sigma = 2$) and fast ($\sigma = 4$) decay rates and finite element discretization degrees $p = 1$ and $p = 3$ on the L-shaped domain.

We observe that there is no significant difference in the error and estimator magnitudes in the experiments between the two different computational domains. For the results on the unit square domain, we refer to Supplement D. The experiments show that the deterministic estimator contribution η_{det} captures the singularity of the L-shaped domain and prioritizes to refine the mesh at the reentrant corner as known from deterministic adaptive FE methods.

The $p = 3$ FE discretizations converge at roughly thrice the rate of the $p = 1$ cases, which is expected. The rates for both FE discretizations are the same as observed in [1, 9, 25, 55] in the affine case. For the lognormal case, to the knowledge of the authors only [3] has presented reliable error estimation yet. Our results are again very similar to the results reported previously.

In both magnitude and convergence rate we observe almost no difference between the two decay rates. The only exception to this is the $p = 3$ FE discretization for the affine case, where the error and estimator for $\sigma = 4$ reached slightly smaller values.

The obtained convergence rates for the respective affine and the lognormal experiments are the same, which indicates that the adaptive algorithm works robustly independent of the specific choice of the diffusion coefficient. We observe that the overall magnitudes of error and estimator are slightly larger in the more involved lognormal case, but the estimator consistently overestimates the error by a factor of approximately 10. We also note that both error and estimator values are of the same order of magnitude with respect to the degrees of freedom as in [1, 3, 9, 25, 55].

The most obvious difference between the two diffusion coefficient types is the smoothness of the numerical convergence graphs. In the affine case the error and error estimator decrease monotonously, whereas they might sometimes increase after refinement in the lognormal case (B). Interestingly, this is most prevalent when considering a fast decay rate of $\sigma = 4$. Nevertheless, this behaviour is expected since the reconstruction accuracy of the forward problem depends on the set of samples that is used in the training. The samples are drawn randomly and we probably do not use sufficiently many samples to satisfy the restricted isometry properties of Theorem 3.1 with high probability. Consequently, the VMC algorithm might not find a strictly better (or equally good) approximation in each step. In particular, we observe that error and estimator might increase after refinement of the spatial mesh $\mathcal{T}$ or enlarging the stochastic approximation space Λ_d when the amount of training samples is not sufficiently increased yet.

We also observe that it is possible for the error and estimator to stagnate for several iterations before continuing convergence. Due to our refinement strategy, which enlarges the deterministic and stochastic spaces via Dörfler marking and the regression samples by a relative factor, the required extensive enlargement of Λ_d and N_{VMC} thus might need multiple iterations to reduce the error significantly.

6.5. **Representation complexity of the solution.** In this section we discuss the memory complexity of the TT representation of the solution for the experiments during the adaptive refinement of Algorithm 3. Figure 2 shows the growth in the number of modes, the stochastic dimensions and the representation ranks of $u_{N,r,n}$ for the affine and lognormal case, respectively.

In the affine case we observe the expected behaviour, namely that for the fast decay $\sigma = 4$ the refinement Algorithm 3 focuses on increasing the maximum degree of the stochastic polynomials whereas in the experiments with slow decay $\sigma = 2$ increasing the number of active modes is prioritized independently on the computational domain. We note that the dimension of the first stochastic parameter is always the largest except for the FE ansatz space dimension. Moreover, the dimensions decrease
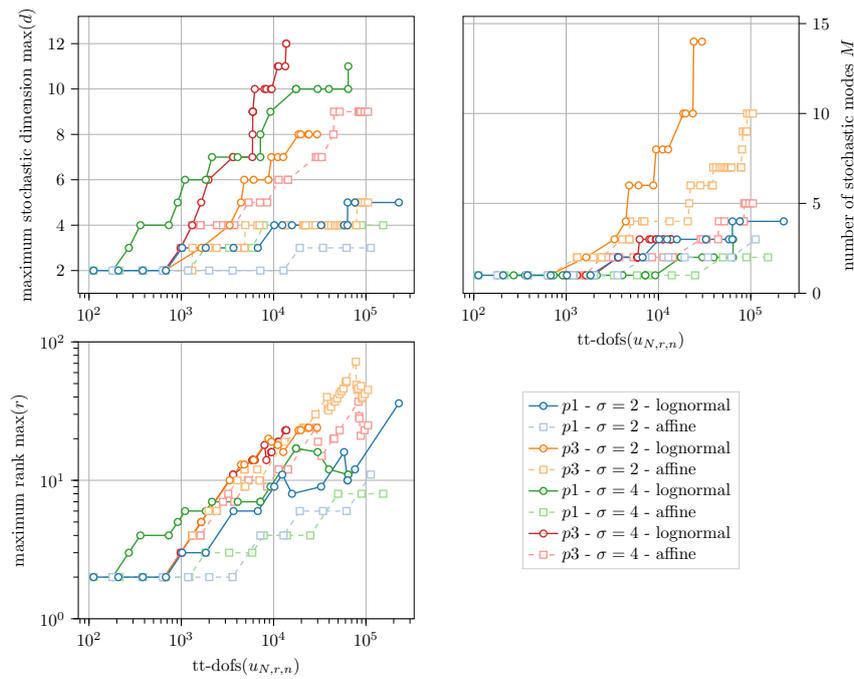
FIGURE 2. Maximum dimensions, ranks and number of active modes of the solution $u_{N,r,n}$ with respect to the tt-dofs of $u_{N,r,n}$ for the affine and lognormal case on the L-shaped domain.

monotonously for larger modes. Algorithm 3 requires two to three times more iterations if cubic basis functions $p = 3$ are used in the FE discretization. This is again expected bahaviour since the mesh $\mathcal{T}$ needs to be refined less often due to the better approximation properties of higher order FE methods. In the bottom left graph of Figure 2 we see that the maximal representation rank of the solution $u_{N,r,n}$ is up to six times larger for the experiments where we let $p = 3$. It seems that the maximal rank increases with an almost constant rate with respect to the tt-dofs$(u_{N,r,n})$. The first rank is almost always the largest and that the ranks decrease for larger stochastic modes.

In the lognormal case, Algorithm 3 behaves as expected as well, i.e. $\sigma = 2$ requires the activation of more stochastic modes and $\sigma = 4$ prioritizes larger polynomial dimensions in the first modes. The overall number of active modes is similar to the affine case and the maximal polynomial degree is slightly larger. We suppose that the latter stems from the more complicated structure of $u_{N,r,n}$ for the lognormal diffusion coefficient κ. The ranks also increase at an overall constant rate with respect to tt-dofs$(u_{N,r,n})$ although the variation of coefficient realizations in case (B) is larger than in case (A). A comparison to [3] shows that the number of active modes and the maximal dimensions and ranks behave similarly in growth rate and magnitude.

Table 1 depicts the number of training samples N_{VMC} used in the last iteration before Algorithm 3 terminates. Most notably we see that the lower order FE discretizations with $p = 1$ require up to two orders of magnitude fewer training samples during the algorithm. Moreover, the number of samples for the affine case with $p = 1$ are not increased during any iteration of Algorithm 3. This can be explained by the high regularity of the solution u of (1.1) and the relatively large spatial error of the low-order FE discretization. Opposite to this, we arrive at 10^5 samples before Algorithm 3 terminates for the $p = 3$ lognormal experiments, which can be explained by the same argument. This also causes the adaptive algorithm to terminate prematurely for fast decay $\sigma = 4$ with $p = 3$ in case (B). Here, Algorithm 1 would require an extensive increase of the number of training samples N_{VMC} that exceeds 10^6. This is set as a limit since reconstruction times for the solution become prohibitive otherwise.

	affine		lognormal	
	$\sigma = 2$	$\sigma = 4$	$\sigma = 2$	$\sigma = 4$
$p = 1$	100	100	1600	5377
$p = 3$	13280	3280	3752	162424

TABLE 1. Number of training samples N_{VMC} used in the last iteration of Algorithm 3 for all experiments on the L-shaped domain.

6.6. **Refinement strategy evaluation.** In this section we take a detailed look at the concrete refinement decisions of Algorithm 2. Figure 3 depicts the error $\mathcal{E}_u(u_{N,r,n})$, the total estimator $\eta(u_{N,r,n})$ and the three estimator contributions $\eta_{\mathrm{det}}(u_{N,r,n})$, $\eta_{\mathrm{sto}}(u_{N,r,n})$ and $\eta_{\mathrm{alg}}(u_{N,r,n})$ exemplarily for the lognormal experiments on the L-shaped domain for $p = 1$ with fast decay $\sigma = 4$ (left) and $p = 3$ with slow decay $\sigma = 2$ (right). The background patterns indicate by color which of the estimator parts dominate in the respective iteration. The background hatches display the quantity that Algorithm 2 chooses to increase based on the dominating estimator contribution and the estimated minimal Gramian eigenvalue. To provide a more detailed view on the stochastic refinement, we distinguish between an increase in the polynomial degrees of the stochastic space and the activation of additional modes.
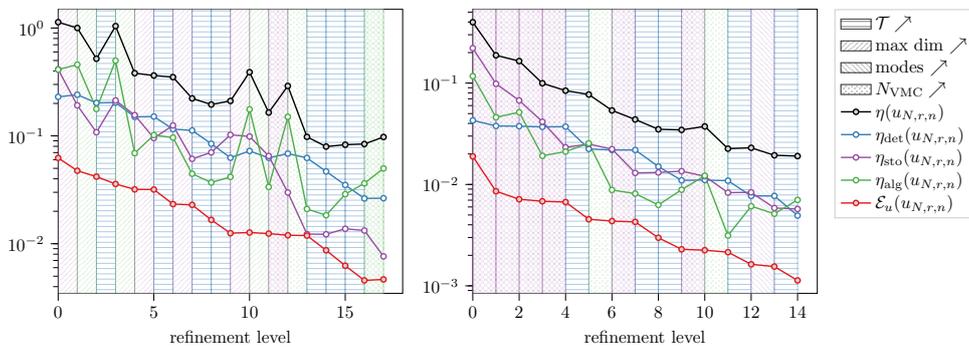

FIGURE 3. Error $\mathcal{E}_u(u_{N,r,n})$, total estimator $\eta(u_{N,r,n})$ and estimator contributions for each refinement level for the lognormal case on the L-shaped domain for $p = 1$ with $\sigma = 4$ (left) and $p = 3$ with $\sigma = 2$ (right). The background displays the dominating estimator contribution (color) and the resulting refinement strategy (hatching). The patterns are superimposed when multiple quantities are refined simultaneously.

In the case of the linear FE space $p = 1$ with slow decay $\sigma = 4$ (Figure 3, left) each estimator contribution dominates at some point, which confirms that the error bound indeed relies on all three of them. We also note that even though $\eta_{\mathrm{alg}}(u_{N,r,n})$ dominates several times in the first few iterations, Algorithm 1 causes an increase of the VMC training samples only after extensively refining the spatial mesh $\mathcal{T}$ and the dimension of the first mode d_1. Due to the fast decay $\sigma = 4$, the number of modes has to be increased only once and only after extensive refinement of the other quantities. In the case of the cubic FE space $p = 3$ with fast decay $\sigma = 2$ (Figure 3, right), the emphasis of Algorithm 2 lies on the refinement of the stochastic space in the first iterations.

REFERENCES

[1] M. Eigel, C. J. Gittelson, C. Schwab, and E. Zander. "Adaptive stochastic Galerkin FEM". In: *Computer Methods in Applied Mechanics and Engineering* 270 (03/2014), pp. 247–269.

[2] M. Eigel, M. Marschall, and R. Schneider. "Sampling-free Bayesian inversion with adaptive hierarchical tensor representations". In: *Inverse Problems* 34.3 (02/2018), p. 035010.

[3] M. Eigel, M. Marschall, M. Pfeffer, and R. Schneider. "Adaptive stochastic Galerkin FEM for lognormal coefficients in hierarchical tensor representations".
In: *Numerische Mathematik* 145.3 (06/2020), pp. 655–692.

[4] M. Eigel, R. Schneider, P. Trunschke, and S. Wolf. "Variational Monte Carlo—bridging concepts of machine learning and high-dimensional partial differential equations".
In: *Advances in Computational Mathematics* 45.5-6 (10/2019), pp. 2503–2532.

[5] M. Eigel, P. Trunschke, and R. Schneider.
"Convergence bounds for empirical nonlinear least-squares". In: (2020).

[6] M. Eigel, N. Farchmin, S. Heidenreich, and P. Trunschke.
Efficient approximation of high-dimensional exponentials by tensornetworks. 2021.
arXiv: 2105.09064 [math.NA].

[7] P. Trunschke.
Convergence bounds for nonlinear least squares and applications to tensor recovery. 2021.
arXiv: 2108.05237 [math.NA].

[8] M. Eigel, C. Gittelson, C. Schwab, and E. Zander. *A convergent adaptive stochastic Galerkin finite element method with quasi-optimal spatial meshes*. Tech. rep. 2014-01.
Switzerland: Seminar for Applied Mathematics, ETH Zürich, 2014.

[9] A. Bespalov, D. Praetorius, L. Rocchi, and M. Ruggeri.
"Convergence of Adaptive Stochastic Galerkin FEM". In: 57.5 (01/2019), pp. 2359–2382.

[10] A. Bespalov, D. Praetorius, and M. Ruggeri.
"Convergence and rate optimality of adaptive multilevel stochastic Galerkin FEM".
In: (05/2021).

[11] C. Carstensen, M. Eigel, R. H. Hoppe, and C. Löbhard.
"A review of unified a posteriori finite element error control".
In: *Numerical Mathematics: Theory, Methods and Applications* 5.4 (2012), pp. 509–558.

[12] C. Schwab and C. J. Gittelson.
"Sparse tensor discretizations of high-dimensional parametric and stochastic PDEs".
In: *Acta Numerica* 20 (2011), pp. 291–467.

[13] A. Cohen and R. DeVore. "Approximation of high-dimensional parametric PDEs".
In: *Acta Numerica* 24 (2015), pp. 1–159.

[14] J. Galvis and M. Sarkis.
"Approximating infinity-dimensional stochastic Darcy's equations without uniform ellipticity".
In: *SIAM J. Numer. Anal.* 47.5 (2009), pp. 3624–3651.

[15] A. Mugler. "Verallgemeinertes polynomielles Chaos zur Lösung stationärer Diffusionsprobleme mit zufälligen Koeffizienten". doctoralthesis. BTU Cottbus - Senftenberg, 2013.

[16] M. Bachmayr, A. Cohen, R. DeVore, and G. Migliorati.
"Sparse polynomial approximation of parametric elliptic PDEs. Part II: lognormal coefficients".
In: *ESAIM: Mathematical Modelling and Numerical Analysis* 51.1 (2017), pp. 341–363.

[17] F. Nobile, R. Tempone, and C. G. Webster. "An Anisotropic Sparse Grid Stochastic Collocation Method for Partial Differential Equations with Random Input Data".
In: 46.5 (01/2008), pp. 2411–2442.

[18] F. Nobile, R. Tempone, and C. G. Webster. "A Sparse Grid Stochastic Collocation Method for Partial Differential Equations with Random Input Data". In: 46.5 (01/2008), pp. 2309–2345.

[19] O. G. Ernst and B. Sprungk.
"Stochastic collocation for elliptic PDEs with random data: the lognormal case".
In: *Sparse Grids and Applications-Munich 2012*. Springer, 2014, pp. 29–53.

[20] O. Le Maitre and O. M. Knio. *Spectral Methods for Uncertainty Quantification: With Applications to Computational Fluid Dynamics (Scientific Computation)*. English. Springer, 04/2010, p. 552.

[21] R. G. Ghanem and P. Spanos. *Stochastic Finite Elements: A Spectral Approach*. English.
Springer, 12/1990, p. 214.

[22]   M. Eigel, C. Merdon, and J. Neumann. "An Adaptive Multilevel Monte Carlo Method with Stochastic Bounds for Quantities of Interest with Uncertain Data".
        In: *SIAM/ASA Journal on Uncertainty Quantification* 4.1 (01/2016), pp. 1219–1245.

[23]   A. Bespalov, C. E. Powell, and D. Silvester.
        "Energy Norm A Posteriori Error Estimation for Parametric Operator Equations".
        In: *SIAM Journal on Scientific Computing* 36.2 (01/2014), A339–A363.

[24]   A. Bespalov and D. Silvester.
        "Efficient Adaptive Stochastic Galerkin Methods for Parametric Operator Equations".
        In: 38.4 (01/2016), A2118–A2140.

[25]   A. Bespalov and L. Rocchi. "Efficient Adaptive Algorithms for Elliptic PDEs with Random Data".
        In: 6.1 (01/2018), pp. 243–272.

[26]   A. J. Crowder, C. E. Powell, and A. Bespalov. "Efficient Adaptive Multilevel Stochastic Galerkin Approximation Using Implicit A Posteriori Error Estimation".
        In: *SIAM Journal on Scientific Computing* 41.3 (01/2019), A1681–A1705.

[27]   C. M. Bryant, S. Prudhomme, and T. Wildey. "Error Decomposition and Adaptivity for Response Surface Approximations from PDEs with Parametric Uncertainty".
        In: 3.1 (01/2015), pp. 1020–1045.

[28]   S. Prudhomme and C. M. Bryant. "Adaptive surrogate modeling for response surface approximations with application to bayesian inference". In: 2.1 (09/2015).

[29]   M. Bachmayr and I. Voulis. "An adaptive stochastic Galerkin method based on multilevel expansions of random fields: Convergence and optimality".
        In: *arXiv preprint arXiv:2109.09136* (2021).

[30]   M. Bachmayr, R. Schneider, and A. Uschmajew. "Tensor Networks and Hierarchical Tensors for the Solution of High-Dimensional Partial Differential Equations".
        In: *Foundations of Computational Mathematics* 16.6 (04/2016), pp. 1423–1472.

[31]   W. Hackbusch and R. Schneider. "Tensor Spaces and Hierarchical Tensor Representations".
        In: *Extraction of Quantifiable Information from Complex Systems*.
        Springer International Publishing, 2014, pp. 237–261.

[32]   A. Nouy. "Chapter 4: Low-Rank Methods for High-Dimensional Approximation and Model Order Reduction". In: Society for Industrial and Applied Mathematics, 07/2017, pp. 171–226.

[33]   I. V. Oseledets and E. E. Tyrtyshnikov.
        "Breaking the Curse of Dimensionality, Or How to Use SVD in Many Dimensions".
        In: *SIAM Journal on Scientific Computing* 31.5 (01/2009), pp. 3744–3759.

[34]   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".
        In: *SIAM/ASA Journal on Uncertainty Quantification* 3.1 (01/2015), pp. 1109–1135.

[35]   S. Dolgov and R. Scheichl.
        "A Hybrid Alternating Least Squares–TT-Cross Algorithm for Parametric PDEs".
        In: *SIAM/ASA Journal on Uncertainty Quantification* 7.1 (01/2019), pp. 260–291.

[36]   S. Dolgov, K. Anaya-Izquierdo, C. Fox, and R. Scheichl. "Approximation and sampling of multivariate probability distributions in the tensor train decomposition".
        In: *Statistics and Computing* 30.3 (11/2019), pp. 603–625.

[37]   M. Eigel, R. Gruhlke, and M. Marschall. *Low-rank tensor reconstruction of concentrated densities with application to Bayesian inversion*. 2020. arXiv: 2008.04264 [math.NA].

[38]   S. Dolgov, B. N. Khoromskij, A. Litvinenko, and H. G. Matthies.
        Computation of the Response Surface in the Tensor Train data format. 2014.
        arXiv: 1406.2816 [math.NA].

[39] M. Eigel, M. Marschall, and M. Multerer.
An adaptive stochastic Galerkin tensor train discretization for randomly perturbed domains.
2019. arXiv: `1902.07753 [math.NA]`.

[40] M. Espig, W. Hackbusch, A. Litvinenko, H. G. Matthies, and P. Wähnert.
"Efficient low-rank approximation of the stochastic Galerkin matrix in tensor formats".
In: *Computers & Mathematics with Applications* 67.4 (2014), pp. 818–829.

[41] I. Oseledets and E. Tyrtyshnikov. "TT-cross approximation for multidimensional arrays".
In: *Linear Algebra and its Applications* 432.1 (01/2010), pp. 70–88.

[42] V. H. Hoang and C. Schwab. "N-term Wiener chaos approximation rates for elliptic PDEs with lognormal gaussian random inputs".
In: *Mathematical Models and Methods in Applied Sciences* 24.04 (01/2014), pp. 797–826.

[43] C. J. Gittelson.
"Stochastic Galerkin Discretization of the Log-Normal isotropic Diffusion Problem".
In: *Mathematical Models and Methods in Applied Sciences* 20.02 (02/2010), pp. 237–263.

[44] A. Mugler and H.-J. Starkloff. "On the convergence of the stochastic Galerkin method for random elliptic partial differential equations".
In: *ESAIM: Mathematical Modelling and Numerical Analysis-Modélisation Mathématique et Analyse Numérique* 47.5 (2013), pp. 1237–1263.

[45] A. Cohen, R. DeVore, and C. Schwab.
"Convergence Rates of Best N-term Galerkin Approximations for a Class of Elliptic sPDEs".
In: *Foundations of Computational Mathematics* 10.6 (07/2010), pp. 615–646.

[46] A. Cohen, R. DeVore, and C. Schwab.
"Analytic regularity and polynomial approximation of parametric and stochastic elliptic PDEs".
In: 09.01 (01/2011), pp. 11–47.

[47] I. V. Oseledets. "Tensor-Train Decomposition".
In: *SIAM Journal on Scientific Computing* 33.5 (01/2011), pp. 2295–2317.

[48] S. Holtz, T. Rohwedder, and R. Schneider. "On manifolds of tensors of fixed TT-rank".
In: *Numerische Mathematik* 120.4 (09/2011), pp. 701–731.

[49] M. Eigel, M. Pfeffer, and R. Schneider.
"Adaptive stochastic Galerkin FEM with hierarchical tensor representations".
In: *Numerische Mathematik* 136.3 (11/2016), pp. 765–803.

[50] M. Eigel, J. Neumann, R. Schneider, and S. Wolf.
"Non-intrusive Tensor Reconstruction for High-Dimensional Random PDEs".
In: *Computational Methods in Applied Mathematics* 19 (2019), pp. 39–53.

[51] A. Cohen and G. Migliorati. "Optimal weighted least-squares methods".
In: *The SMAI journal of computational mathematics* 3 (2017), pp. 181–203.

[52] R. Stevenson. "The completion of locally refined simplicial partitions created by bisection".
In: *Mathematics of Computation* 77.261 (2008), pp. 227–241.

[53] B. Huber and S. Wolf. *Xerus - A General Purpose Tensor Library.*
`https://libxerus.org/`. 2014–2021.

[54] M. Alnæs, J. Blechta, J. Hake, A. Johansson, B. Kehlet, A. Logg, C. Richardson, J. Ring, M. E. Rognes, and G. N. Wells. "The FEniCS Project Version 1.5". en.
In: *Archive of Numerical Software* Vol 3 (2015), pp. 9–23.

[55] A. Bespalov, D. Praetorius, and M. Ruggeri.
Two-level a posteriori error estimation for adaptive multilevel stochastic Galerkin FEM. 2021.
arXiv: `2006.02255 [math.NA]`.

[56] P. B. Rohrbach, S. Dolgov, L. Grasedyck, and R. Scheichl.
"Rank bounds for approximating gaussian densities in the tensor-train format".
In: *arXiv preprint arXiv:2001.08187* (2020).

[57] L. Grasedyck, D. Kressner, and C. Tobler.
      "A literature survey of low-rank tensor approximation techniques".
      In: *GAMM-Mitteilungen* 36.1 (08/2013), pp. 53–78.
[58] B. N. Khoromskij. "Tensor numerical methods for multidimensional PDES: theoretical analysis
      and initial applications". In: *ESAIM: Proceedings and Surveys* 48 (01/2015). Ed. by
      N. Champagnat, T. Lelièvre, and A. Nouy, pp. 1–28.
[59] T. G. Kolda and B. W. Bader. "Tensor Decompositions and Applications".
      In: *SIAM Review* 51.3 (08/2009), pp. 455–500.
[60] W. Hackbusch. *Tensor Spaces and Numerical Tensor Calculus (Springer Series in
      Computational Mathematics Book 42)*. English. Springer, 02/2012, p. 524.
[61] L. Grasedyck. "Hierarchical Singular Value Decomposition of Tensors".
      In: *SIAM Journal on Matrix Analysis and Applications* 31.4 (01/2010), pp. 2029–2054.
[62] M. Loeve. *Probability Theory II (Graduate Texts in Mathematics, 46)*. Springer, 05/1978, p. 432.

APPENDIX A. ORTHOGONAL POLYNOMIAL BASIS FUNCTIONS

Legendre polynomials. The standard Legendre polynomials $\tilde{L}_j$ of degree $j \in \mathbb{N}_0$ constitute an orthogonal basis of $L^2(\Gamma_\ell, \pi_\ell)$ for $\Gamma_\ell = [-1, 1]$ and $\pi_\ell(y_\ell) \equiv 1/2 \, \mathrm{d}y_\ell$. With $c_j = \sqrt{2j+1}$ the set of polynomials $\{L_j = c_j \tilde{L}_j\}_{j=0}^{\infty}$ is orthogonal and normalized with respect to $L^2(\Gamma_\ell, \pi_\ell)$. To derive an analytical expression of the triple product $\tau_{ijk} = \mathbb{E}_{\pi_\ell}[L_i L_j L_k]$ of the normalized Legendre polynomials for any $i, j, k \in \mathbb{N}_0$, we define

$$A_{\mathrm{Leg}}(n) = (2n)! \, 2^{-n} (n!)^{-2}.$$

The triple product τ_{ijk} is then given by

$$(\text{A.1}) \qquad \tau_{ijk} = \begin{cases} \frac{c_i c_j c_k}{(2s+1)} \frac{A_{\mathrm{Leg}}(s-i) A_{\mathrm{Leg}}(s-j) A_{\mathrm{Leg}}(s-k)}{A_{\mathrm{Leg}}(s)} & \text{if } s \in \mathbb{N}_0 \text{ and } s \geqslant \max\{i, j, k\}, \\ 0 & \text{else}, \end{cases}$$

where $s = (i + j + k)/2$.

Hermite polynomials. The standard probabilists Hermite polynomials $\tilde{H}_j$ of degree $j \in \mathbb{N}_0$ constitute an orthogonal basis of $L^2(\Gamma_\ell, \pi_\ell)$ for $\Gamma_\ell = \mathbb{R}$ and $\pi_\ell(y_\ell) = (2\pi)^{-1/2} \exp(-y_\ell^2/2) \, \mathrm{d}y_\ell$. With $c_j = 1/\sqrt{j!}$ the set of polynomials $\{H_j = c_j \tilde{H}_j\}_{j=0}^{\infty}$ is orthogonal and normalized with respect to $L^2(\Gamma_\ell, \pi_\ell)$. To derive an analytical expression of the triple product $\tau_{ijk} = \mathbb{E}_{\pi_\ell}[H_i H_j H_k]$ of the normalized Hermite polynomials for any $i, j, k \in \mathbb{N}_0$, we define

$$A_{\mathrm{Her}}(i, j, k) = \left(\frac{i+j+k}{2}\right)!.$$

With the condition

$$(\text{A.2}) \qquad \frac{1}{2}(i + j + k) \in \mathbb{N}_0 \quad \text{and} \quad \frac{1}{2}(i + j + k) \geqslant \max\{i, j, k\},$$

the triple product τ_{ijk} is given by

$$(\text{A.3}) \qquad \tau_{ijk} = \begin{cases} c_i c_j c_k \frac{i! j! k!}{A_{\mathrm{Her}}(i,j,-k) A_{\mathrm{Her}}(i,-j,k) A_{\mathrm{Her}}(-i,j,k)} & \text{if } i, j, k \text{ satisfy (A.2)}, \\ 0 & \text{else}. \end{cases}$$

Scaled Hermite polynomials. For Gaussian distributions $\mathcal{N}(0, \sigma^2)$ with $\sigma \neq 1$ it is easy to scale the standard probabilists Hermite polynomials to obtain an orthogonal basis. Recall from Section 2 that the univariate density for $\mathcal{N}(0, \sigma_\ell(\rho)^2)$ is given by

$$\pi_\ell(y_\ell; \sigma_\ell(\rho)) = \zeta_\ell(y_\ell, \sigma_\ell(\rho)) \, \pi_\ell(y_\ell, 1)$$

for $\sigma_\ell(\rho) = \exp(\rho\|\gamma_\ell\|_{L^\infty(D)})$ and $\rho \geqslant 0$. Note that $\sigma(\rho) = (\sigma_1(\rho), \sigma_2(\rho), \dots) \in \exp(\ell^1(\mathbb{N}))$ and define the transformation

$$\iota_{\sigma(\rho)} \colon \mathbb{R}^\infty \to \mathbb{R}^\infty, \qquad (y_\ell)_{\ell \in \mathbb{N}} \mapsto (\sigma_\ell(\rho)^{-1} y_\ell)_{\ell \in \mathbb{N}}.$$

With this we can define the multivariate scaled Hermite polynomials $H_\mu^\rho := H_\mu \circ \iota_{\sigma(\rho)}$, where H_μ is the multivariate and normalized standard probabilists Hermite polynomial. The set $\{H_\mu^\rho\}_{\mu \in \mathcal{F}}$ thus forms an orthogonal and normalized basis of $L^2(\Gamma_\kappa, \pi_\rho)$. Moreover, for any $w \in L^2(\Gamma_\kappa, \pi_\rho)$ it holds

$$\int_{\Gamma_\kappa} w \, \mathrm{d}\pi_0(y) = \int_{\Gamma_\kappa} w \circ \iota_{\sigma(\rho)} \, \mathrm{d}\pi_\rho,$$

which implies that $\mathbb{E}_{\pi_\rho}[H_i^\rho H_j^\rho H_k^\rho] = \mathbb{E}_{\pi_0}[H_i H_j H_k]$ for any $i, j, k \in \mathbb{N}_0$. Hence, the triple product with respect to the scaled Hermite polynomials can be computed by (A.3) as well.

APPENDIX B. WELL-POSEDNESS OF THE LOGNORMAL CASE

First we note that by [42, Lemma 2.1] the set Γ_κ from (2.8) is measurable and it holds $\pi_0(\Gamma_\kappa) = 1$. Moreover, Lemma 2.2 of [42] shows that the lognormal field κ is bounded and positive. However, boundedness and positivity only hold pointwise and not uniformly over the parameter space Γ_κ. Even though this leads to a far more intricate analysis, following the arguments of [12] it is still possible to obtain a well-defined variational formulation by the introduction of the stronger measure $\pi_{\vartheta\rho}$ from (2.9) for some $\rho > 0$ and $0 < \vartheta < 1$. Recall that the bilinear form in the lognormal case (B) is given by

$$B_{\vartheta\rho}(w, v) = \int_{\Gamma_\kappa} \int_D \kappa \, \nabla w \cdot \nabla v \, \mathrm{d}x \, \mathrm{d}\pi_{\vartheta\rho}(y)$$

and that the solution space is defined via

$$\mathcal{V}_{\vartheta\rho} = \{w \colon \Gamma_\kappa \to \mathcal{X} \text{ measurable with } B_{\vartheta\rho}(w, w) < \infty\}.$$

By [12, Proposition 2.43] it then follows that

$$L^2(\Gamma_\kappa, \pi_\rho; \mathcal{X}) \subset \mathcal{V}_{\vartheta\rho} \subset L^2(\Gamma_\kappa, \pi_0; \mathcal{X}) \qquad \text{for any } 0 < \vartheta < 1$$

are continuous embeddings. With this, Lemma 2.41 and Lemma 2.42 from [12] show that the bilinear form $B_{\vartheta\rho}$ is $\mathcal{V}_{\vartheta\rho}$–elliptic and bounded in the sense that

(B.1) $\quad |B_{\vartheta\rho}(w, v)| \leqslant \hat{c}(\vartheta\rho)\|w\|_{L^2(\Gamma_\kappa, \pi_\rho; \mathcal{X})}\|v\|_{L^2(\Gamma_\kappa, \pi_\rho; \mathcal{X})} \qquad \text{for all } w, v \in L^2(\Gamma_\kappa, \pi_\rho; \mathcal{X}),$

(B.2) $\quad B_{\vartheta\rho}(w, w) \geqslant \check{c}(\vartheta\rho)\|w\|_{L^2(\Gamma_\kappa, \pi_0; \mathcal{X})}^2 \qquad\qquad \text{for all } w \in L^2(\Gamma_\kappa, \pi_0; \mathcal{X}).$

APPENDIX C. THE TENSOR TRAIN FORMAT

In the following we briefly recall the TT format and some fundamental properties of TTs. For a more detailed overview, we refer the reader to [30, 33, 48, 49, 56] and the references therein. Any function $w_N \in \mathcal{V}_N(\Lambda_d; \mathcal{T}, p)$ has an expansion of the form

(C.1) $\quad w_N(x, y) = \sum_{j \in [|\mathcal{T}|]} \sum_{\mu \in \Lambda_d} \boldsymbol{w}[j, \mu]\varphi_j(x)P_\mu(y) \qquad \text{with} \qquad \boldsymbol{w} \in \mathbb{R}^{|\mathcal{T}| \times d_1 \times \cdots \times d_M}.$

Hence, $\mathcal{V}_N$ is isomorphic to the space of coefficient tensors $\mathbb{R}^{|\mathcal{T}| \times d} = \mathbb{R}^{|\mathcal{T}| \times d_1 \times \cdots \times d_M}$. Note that the size of the coefficient tensor $\boldsymbol{w}$ grows exponentially with the order M, which is commonly referred to as the *curse of dimensionality*. To mitigate this exponential dependence on M, we employ a low-rank decomposition of the tensor $\boldsymbol{w}$. There are many tensor decompositions available [57–60], but due to

its simplicity and the wide availability in numerical libraries we chose the TT format for our derivations. The TT representation of a tensor $\boldsymbol{w} \in \mathbb{R}^{J \times d}$, $J \in \mathbb{N}$, is given by

$$(C.2) \quad \boldsymbol{w}[j, \mu] = \sum_{k_1=1}^{r_1} \ldots \sum_{k_M=1}^{r_M} \boldsymbol{w}_0[j, k_1] \prod_{m=1}^{M} \boldsymbol{w}_m[k_m, \mu_m, k_{m+1}] \quad \text{for any } j \in [J] \text{ and } \mu \in \Lambda_d$$

$$(C.3) \qquad =: \sum_{k=1}^{r} \boldsymbol{w}_0[j, k_1] \prod_{m=1}^{M} \boldsymbol{w}_m[k_m, \mu_m, k_{m+1}]$$

for some $r \in \mathbb{N}^M$, where we use the convention $r_{M+1} = 1$. In our application the zeroth component tensor $\boldsymbol{w}_0 \in \mathbb{R}^{J \times r_1}$ corresponds to the spatial discretization. The stochastic components are given as order three tensors $\boldsymbol{w}_m \in \mathbb{R}^{r_m \times d_m \times r_{m+1}}$. If all ranks r_m are minimal, this is called *tensor train decomposition* of $\boldsymbol{w}$ with TT rank r and we write $\text{tt-rank}(\boldsymbol{w}) = r$. The set of all tensors of TT rank r forms a manifold [48] of dimension

$$\text{tt-dofs}(\boldsymbol{w}) = Jr_1 - r_1^2 + \sum_{m=1}^{M-1} \left(r_m d_m r_{m+1} - r_{m+1}^2 \right) + r_M d,$$

which shows that the complexity of the TT format behaves like $\mathcal{O}(J\hat{r} + M\hat{d}\hat{r}^2)$ for $\hat{d} = \max\{d_1, \ldots, d_M\}$ and $\hat{r} = \max\{r_1, \ldots, r_M\}$. In contrast to full tensor representations, with complexity $\mathcal{O}(J\hat{d}^M)$, TTs depend only linearly on the order M. As a result, the TT format is especially efficient for a small maximal rank $\hat{r}$.

Similarly we can express linear operators $W : \mathcal{V}_N(\Lambda_d; \mathcal{T}, p) \to \mathcal{V}_N(\Lambda_q; \mathcal{T}, p)$ in the TT format. For this recall that the application of W to $v_N \in \mathcal{V}_N(\Lambda_d; \mathcal{T}, p)$ yields

$$Wv_N(x, y) = \sum_{i \in [N]} \sum_{\nu \in \Lambda_q} \sum_{j \in [N]} \sum_{\mu \in \Lambda_d} \boldsymbol{W}[i, \nu; j, \mu] \boldsymbol{v}[j, \mu] \phi_i(x) P_\nu(y).$$

The TT representation of the tensor operator $\boldsymbol{W} : \mathbb{R}^{J \times d} \to \mathbb{R}^{J \times q}$ is thus determined by

$$\boldsymbol{W}[i, \nu; j, \mu] = \sum_{k=1}^{r} \boldsymbol{W}_0[i, j, k_1] \prod_{m=1}^{M} \boldsymbol{W}_m[k_m, \nu_m, \mu_m, k_{m+1}]$$

for any $i, j \in [J]$, $\mu \in \Lambda_d$ and $\nu \in \Lambda_q$ with component tensors $\boldsymbol{W}_0 \in \mathbb{R}^{J \times J \times r_1}$ and $\boldsymbol{W}_m \in \mathbb{R}^{r_m \times q_m \times d_m \times r_{m+1}}$. The TT decomposition always exists and can be computed using the hierarchical singular value decomposition (SVD) [48]. A truncated hierarchical SVD leads to quasi-optimal approximations of the TT decomposition in the Frobenius norm [31, 33, 61, 62]. This can also be applied to tensors which are already given in the TT format to obtain a TT decomposition with a lower rank. This process is referred to as *rounding*.
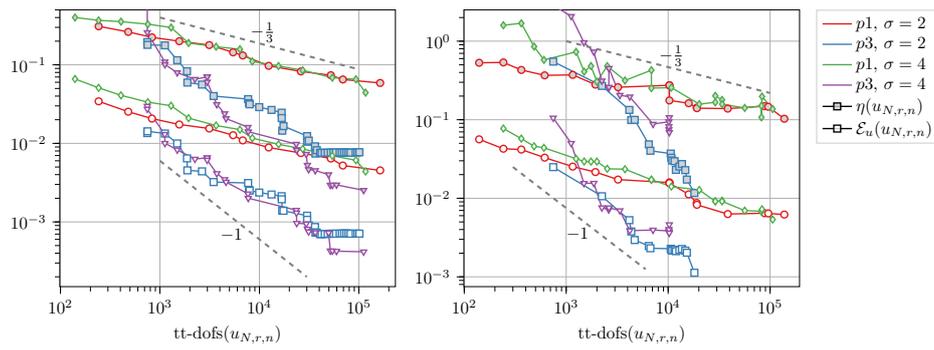
FIGURE 4. Sampled root mean square error $\mathcal{E}_u(u_{N,r,n})$ and total error estimator $\eta(u_{N,r,n})$ of the fully adaptive algorithm on the unit square domain. Considered are finite element approximations of order $p = 1$ and $p = 3$ for slow ($\sigma = 2$) and fast ($\sigma = 4$) decay for the affine (left) and lognormal (right) case.
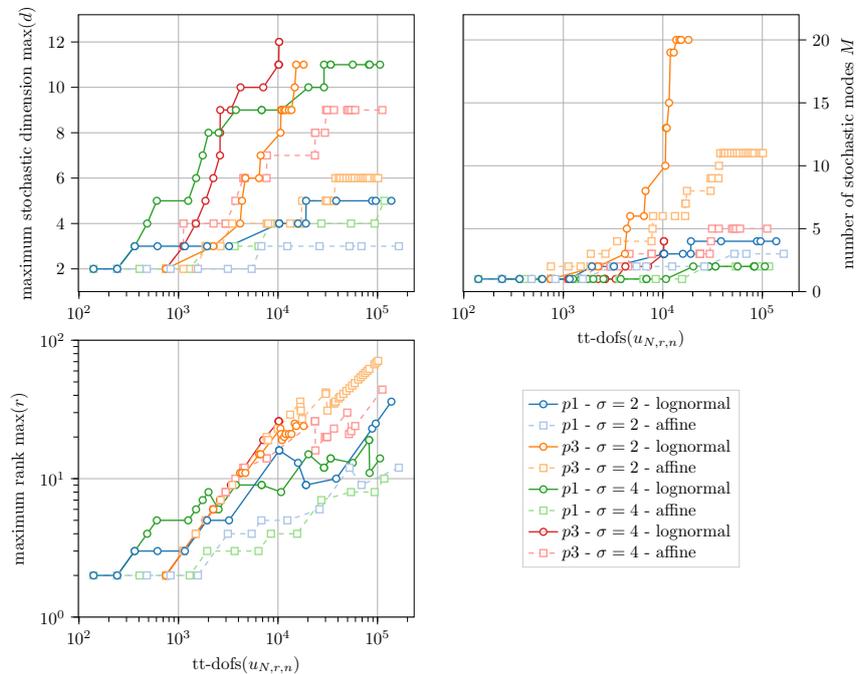

FIGURE 5. Maximum dimensions, ranks and number of active modes of the solution $u_{N,r,n}$ with respect to the tt-dofs of $u_{N,r,n}$ for the affine and lognormal case on the unit square domain.