ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

MATHICSE

Mathematics Institute of Computational Science and Engineering

School of Basic Sciences - Section of Mathematics

MATHICSE Technical Report

Nr. 39.2014

September 2014

Tensor train approximation of moment equations for the log-normal Darcy problem

Francesca Bonizzoni, Fabio Nobile, Daniel Kressner

http://mathicse.epfl.ch

Phone: +41 21 69 37648

Address:
EPFL - SB - MATHICSE (Bâtiment MA)
Station 8 - CH-1015 - Lausanne - Switzerland

Fax: +41 21 69 32545

TENSOR TRAIN APPROXIMATION OF MOMENT EQUATIONS FOR THE LOG-NORMAL DARCY PROBLEM*

FRANCESCA BONIZZONI[†], FABIO NOBILE[‡], AND DANIEL KRESSNER[§]

Abstract. We study the Darcy problem with log-normal permeability, modeling the fluid flow in a heterogeneous porous medium. A perturbation approach is adopted, expanding the solution in Taylor series around the nominal value of the permeability. The resulting recursive deterministic problem satisfied by the expected value of the stochastic solution, analytically derived and studied in [4], is discretized with a full tensor product finite element technique. To overcome the incurred curse of dimensionality the solution is sought in a low-rank tensor format, the so called Tensor Train format. We develop an algorithm for solving the recursive first moment problem approximately in the Tensor Train format and show its effectiveness with numerical examples.

Key words. Uncertainty quantification, Elliptic PDE with random coefficient, Log-normal distribution, Perturbation technique, Moment equations, Low rank approximation.

AMS subject classifications. 65C20, 35R60, 60H25, 35B20, 41A58, 65N30, 65F99, 15A69.

1. Introduction.

In this work we focus on the so called Darcy problem modeling the single-phase fluid flow in a heterogeneous porous medium $D \subset \mathbb{R}^d$, $d = 1, 2, 3$:

$$-div\,(a(\omega, x)\nabla u(\omega, x)) = f(x), \quad x \in D,$$

where $\omega \in \Omega$ denotes an elementary event in a suitable probability space. We describe the uncertain permeability field $a(\omega, x)$ as a *log-normal random field*, that is, $a(\omega, x) = e^{Y(\omega,x)}$ where $Y(\omega, x)$ is a Gaussian random field. The log-normal model is frequently used in geophysical and hydrological applications; see, e.g., [17, 18, 27, 30, 2, 13]. Recently, it has also been discussed in the mathematical literature; see [8, 15, 14].

Given complete statistical information on the Gaussian random field Y, the aim of this paper is to compute the expected value $\mathbb{E}[u]$ of the random solution $u(\omega, x)$ or other statistics of the solution.

In this work, under the assumption of *small variability* of the random field Y, we adopt a *perturbation approach* based on the Taylor expansion of the random solution $u(Y) : L^\infty(D) \to H^1(D)$ with respect to Y, and approximate the expected value of u by the expected value of its K-th degree Taylor polynomial: $\mathbb{E}[u(Y, x)] \simeq \mathbb{E}\left[T^K u(Y, x)\right] = \sum_{k=0}^{K} \frac{\mathbb{E}\left[u^k\right]}{k!}$, where u^k denotes the k-th order Gateaux derivative of u with respect to Y.

In our previous works [5, 4] we have studied the approximation properties of the Taylor polynomial, showing that $T^K u$ converges to u in a bounded open ball of $L^\infty(D)$ with sufficiently small radius, but in general, $\mathbb{E}\left[T^K u\right]$ *does not* converge to $\mathbb{E}[u]$ even for very small variances of Y. We point out the existence of an optimal degree K^σ_{opt} (depending on the standard deviation σ of Y) such that adding further

[*] This work has been supported by the Italian grant FIRB-IDEAS (Project n. RBID08223Z) "Advanced numerical techniques for uncertainty quantification in engineering and life science problems". The authors also acknowledge the support of CADMOS (Center of Advanced Modeling Science).

[†] Faculty of Mathematics, University of Vienna, Oskar-Morgenstern-Platz 1, 1090 Wien, Austria. (francesca.bonizzoni@univie.ac.at)

[‡] CSQI – Calcul Scientifique et Quantification de l'Incertitude, MATHICSE, École Polytechnique Fédérale de Lausanne, Station 8, CH-1015 Lausanne, Switzerland. (fabio.nobile@epfl.ch)

[§] ANCHP – Numerical Algorithms and HPC, MATHICSE, École Polytechnique Fédérale de Lausanne, Station 8, CH-1015 Lausanne, Switzerland. (daniel.kressner@epfl.ch)

terms to the Taylor polynomial will degenerate the accuracy instead of improving it. Nevertheless, for small variances and small order K, the perturbation approach still provides a good approximation of $\mathbb{E}\left[u\right]$.

One possible technique to compute $\mathbb{E}\left[T^K u(Y,x)\right]$ is to expand the random field Y in series, e.g., Fourier or Karhunen-Loéve (KL) expansion, and then truncate it. In this finite-dimensional setting, the Taylor (multivariate) polynomial can be explicitly computed. However, this method suffers from the curse of dimensionality as $\begin{pmatrix} N+K \\ K \end{pmatrix}$ terms have to be computed if N variables are retained in the series expansion of the Gaussian field Y. Note that N can become fairly large when rough fields are considered. Adaptive algorithms have been recently proposed [11, 12], however in the simpler case of a permeability coefficient expanded in a series of uniform random variables, in which case $\mathbb{E}\left[T^K u\right]$ does indeed converge to $\mathbb{E}\left[u\right]$ for sufficiently small variance.

In this work we follow a different path and solve the so called "moment equations", that is, the deterministic equations satisfied by $\mathbb{E}\left[u^k\right]$ for $k \geq 0$. The moment equation technique entails the solution of *recursive* high-dimensional boundary value problems to compute the high order correlations between the random field Y and the derivatives of the random solution u. The perturbation technique coupled with the moment equations has already been used in the literature. In particular, we refer to [32], where the permeability coefficient is modeled as a linear combination of countable many bounded random variables, and to [10], where the authors consider a first order approximation of the m-th moment problem ($m \geq 1$).

To discretize the moment equations, a full tensor product finite element method is used, so that the high-dimensional correlations are represented by high-dimensional tensors. To tackle the curse of dimensionality, we perform all computations and store the results in a *low-rank format*. Low-rank formats aim at representing high-dimensional tensors approximately with a dramatically reduced number of parameters. Recently, two formats based on the singular value decomposition for matrices have been proposed: the Hierarchical Tucker (HT) format [16, 19, 20] and the Tensor Train (TT) format [19, 26]. Both formats can be easily manipulated and express a given tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$ of order d in terms of contractions of d third-order tensors (*cores*) of size $(r_{l-1} \times n_l \times r_l)$, $l = 1, \ldots, d$. The d-tuple $(r_1, \ldots, r_d)$ is called the *rank* of the low rank representation of $\mathcal{X}$.

As an alternative to a full tensor product approach and low rank formats, we mention [9], where results concerning the construction and approximation properties of sparse tensor product polynomial subspaces of Sobolev spaces with bounded mixed derivatives are derived.

In this work we develop an algorithm based on the TT-format for solving the recursive moment equations and computing the K-th order approximation $\mathbb{E}\left[T^K u(Y,x)\right]$. The algorithm requires to access the global and local (element-wise) stiffness matrices corresponding to the mean as well as a TT-representation of the k-points ($k = 1, 2, \ldots, K$) correlations of the input Gaussian random field Y (see [23]). Eventually, everything comes down to the solution of a sequence of deterministic problems with the same stiffness matrix and multiple right hand sides.

If Y is a smooth random field (e.g. with squared exponential covariance function), then it can be well approximated using a number of random variables N sensibly smaller than the number of elements N_h (intervals/triangles/tetrahedra) used in the spacial finite element discretization. On the other hand, if Y is a rough Gaussian random field (e.g. with exponential covariance function) $N = N_h$ random variables are needed to properly catch the behavior of the random field. This fact reflects onto the size of the ranks $(r_1, \ldots, r_k)$ of the TT-representation of the k-points correlation of Y. In [23] the authors prove the theoretical upper bound $r_l \sim N^l$. However, the implementation provided in [31] and described in [23] offers the possibility to dramatically reduce the ranks r_l, and allows to handle high order correlations at least when Y is a smooth random field. For rough random fields, the main limiting factor of our approach for high order correlations is the effectiveness of the low rank representations.

In our computations, the total error is the sum of two contributions: the truncation error of the Taylor series and the tolerance used during the approximate computations and recompressions within the TT-format. When these two errors are properly balanced, we have numerically verified that our approach performs much better than a standard Monte Carlo method, in terms of the number of deterministic problems solved for a given accuracy level.

The outline of the paper is as follows. Section 2 introduces the problem at hand and recalls existing results concerning the well-posedness of the log-normal Darcy problem and approximations via Taylor polynomials. In Section 3 we analytically derive the recursive problem solved by $\mathbb{E}\left[T^K u(Y,x)\right]$, and the finite element formulation of the recursive problem is derived in Section 4. Section 5 introduces the Tensor Train format and in Section 6 the recursive algorithm in Tensor Train format is described. Section 7 studies the storage requirements of the algorithm we have developed. In Section 8 some one-dimensional numerical tests are performed. Conclusions are given in Section 9.

2. Problem setting.

Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a complete probability space, where Ω is the set of outcomes, $\mathcal{F}$ the σ-algebra of events and $\mathbb{P} : \Omega \to [0,1]$ a probability measure. Let D be an open bounded domain in $\mathbb{R}^d$ $(d = 1,2,3)$ with locally Lipschitz boundary.

We study the Darcy boundary value problem with permeability modeled as a log-normal random field: Given a centered (w.l.o.g) Gaussian random field $Y : D\times\Omega \to \mathbb{R}$, $f \in L^2(D)$ and $g \in H^{1/2}(\Gamma_D)$, find $u \in L^p\left(\Omega; H^1(D)\right)$, $p \geq 1$, s.t. $u|_{\Gamma_D} = g$ a.s. and

$$\int_D e^{Y(\omega,x)} \nabla u(\omega,x) \cdot \nabla v(x)\, \mathrm{d}x = \int_D f(x)v(x)\, \mathrm{d}x \quad \forall v \in H^1_{\Gamma_D}(D), \text{ a.s. in } \Omega \tag{2.1}$$

where $\{\Gamma_D,\ \Gamma_N\}$ is a partition of the boundary of the domain ∂D, and homogeneous Neumann boundary conditions are imposed on Γ_N. The limit situation $\Gamma_D = \emptyset$ is also admissible: in this case the solution will be unique up to a constant.

If the covariance function of the random field Y is Hölder regular, $\mathrm{Cov}_Y \in \mathcal{C}^{0,t}(\overline{D \times D})$ for some $0 < t \leq 1$, then there exists a version of Y whose trajectories belong to $\mathcal{C}^{0,\alpha}(\bar{D})$ a.s. for $0 < \alpha < t/2$ (see [4, 5]), and problem (2.1) is well-posed (see [8]). This setting covers also the relevant case in applications of a permeability field conditioned to point-wise measurements. Indeed, given an unconditioned random field Y with Hölder continuous covariance function of exponent t, then the random field Y_{cond} conditioned to available measurements has a covariance function that is Hölder continuous with the same exponent.

Let us define $\sigma^2 = \dfrac{1}{|D|}\displaystyle\int_D \mathbb{V}ar\left[Y(\omega,x)\right] \mathrm{d}x$, which coincides with the variance of the field Y in the case of a stationary field. By abuse of notation, we refer to σ as the standard deviation of Y also in the case when Y is non-stationary.

Under the assumption of small standard deviation $0 < \sigma < 1$, we perform a *perturbation analysis*: we view the stochastic solution u as a map $u : L^\infty(D) \to H^1(D)$ implicitly defined by (2.1), and approximate its expected value by the expected value of its Taylor polynomial of degree K

$$\mathbb{E}\left[u(Y,x)\right] \approx \mathbb{E}\left[T^K u(Y,x)\right] = \sum_{k=0}^{K} \frac{\mathbb{E}\left[u^k\right]}{k!}, \tag{2.2}$$

where u^k is the k-th order Gateaux derivative of u with respect to Y. We refer to $\mathbb{E}\left[u^k\right]$ as the k-th order correction of the mean of u and to $\mathbb{E}\left[T^K u(Y,x)\right]$ as the K-th order approximation of the mean of u.

3. Recursion on the correlations: analytical derivation.

In this section we recall the main steps in the analytical derivation of the recursion satisfied by the corrections of $\mathbb{E}\left[u\right]$ in increasing order. We refer to [4] for details.

The correction of order 0, u^0, is deterministic and is given by the unique solution of the following problem: Given $f \in L^2(D)$ and $g \in H^{1/2}(\Gamma_D)$, find $u^0 \in H^1(D)$ such that $u^0 = g$ on Γ_D and

$$\int_D \nabla u^0(x) \cdot \nabla v(x) \mathrm{d}x = \int_D f(x) v(x) \mathrm{d}x, \quad \forall v \in H^1_{\Gamma_D}(D). \tag{3.1}$$

For $k \geq 1$, the k-th order correction $\mathbb{E}\left[u^k\right]$ satisfies

$$\begin{aligned} &\int_D \nabla \mathbb{E}\left[u^k\right](x) \cdot \nabla v(x)\ \mathrm{d}x \\ &\quad = -\sum_{l=1}^{k} \begin{pmatrix} k \\ l \end{pmatrix} \int_D \mathbb{E}\left[Y^l \nabla u^{k-l}\right](x) \cdot \nabla v(x)\ \mathrm{d}x \quad \forall v \in H^1_{\Gamma_D}(D). \end{aligned} \tag{3.2}$$

To derive a recursion, we first note that the diagonal of $\mathbb{E}\left[Y^{\otimes l} \otimes \nabla u^{k-l}\right]$ on the tensorized domain $D^{\times(l+1)}$ contains the term $\mathbb{E}\left[Y^l \nabla u^{k-l}\right]$ on the right-hand side of (3.2). The following definition allows us to formalize this statement.

Definition 3.1. *Consider a function of n variables $v(x_1, \dots, x_p, \dots, x_q, \dots, x_n)$, for positive integers $1 \leq p \leq q \leq n$. Then the trace function $\mathrm{Tr}_{|_{p:q}} v$ is defined as*

$$\left(\mathrm{Tr}_{|_{p:q}}\right) v(x_1, \dots, x_p, x_{q+1}, \dots, x_n) := v(x_1, \dots, x_{p-1}, \underbrace{x_p, \dots, x_p}_{(q-p+1)-times}, x_{q+1}, \dots, x_n).$$

In particular, Definition 3.1 yields $\mathbb{E}\left[Y^l \nabla u^{k-l}\right] = \mathrm{Tr}_{|_{1:l+1}} \mathbb{E}\left[Y^{\otimes l} \otimes \nabla u^{k-l}\right]$.

We have $\mathbb{E}\left[Y^{\otimes l} \otimes \nabla u^{k-l}\right] = \left(\mathrm{Id}^{\otimes l} \otimes \nabla\right)\ \mathbb{E}\left[Y^{\otimes l} \otimes u^{k-l}\right]$, where $\mathbb{E}\left[Y^{\otimes l} \otimes u^{k-l}\right]$ is the $(l+1)$-point correlation function defined as

$$\mathbb{E}\left[Y^{\otimes l} \otimes u^{k-l}\right](x_1, x_2 \dots, x_{l+1}) := \int_\Omega Y(\omega, x_1) \otimes \dots \otimes Y(\omega, x_l) \otimes u^{k-l}(\omega, x_{l+1}) \mathrm{d}\mathbb{P}(\omega)$$

and the linear operator $\mathrm{Id}^{\otimes l} \otimes \nabla$ applies the gradient operator to the last variable x_{l+1} and the identity operator to all other variables. As a consequence, the k-th order correction $\mathbb{E}\left[u^k\right]$ can be obtained from the $(l+1)$-point correlation functions $\mathbb{E}\left[Y^{\otimes l} \otimes u^{k-l}\right]$ for $l = 1, \dots, k$. The correlation functions themselves satisfy the following recursion.

Recursion on the correlations

(3.3)

$$\boxed{\begin{aligned}
&\text{Given all lower order terms}\\
&\qquad\qquad \mathbb{E}\left[Y^{\otimes(s+l)}\otimes u^{k-l-s}\right]\in\left(L^2(D)\right)^{\otimes(s+l)}\otimes H^1_{\Gamma_D}(D)\text{ for } s=1,\dots,k-l,\\
&\text{find } \mathbb{E}\left[Y^{\otimes l}\otimes u^{k-l}\right]\in\left(L^2(D)\right)^{\otimes l}\otimes H^1_{\Gamma_D}(D)\text{ s.t.}\\
&\int_{D^{\times(l+1)}}\left(\mathrm{Id}^{\otimes l}\otimes\nabla\right)\mathbb{E}\left[Y^{\otimes l}\otimes u^{k-l}\right]\cdot\left(\mathrm{Id}^{\otimes l}\otimes\nabla\right)v\,\mathrm{d}x_1\dots\mathrm{d}x_{l+1}\\
&=-\sum_{s=1}^{k-l}\begin{pmatrix}k-l\\ s\end{pmatrix}\int_{D^{\times(l+1)}}\mathrm{Tr}_{|l+1:l+s+1}\mathbb{E}\left[Y^{\otimes(s+l)}\otimes\nabla u^{k-l-s}\right]\cdot\left(\mathrm{Id}^{\otimes l}\otimes\nabla\right)v\,\mathrm{d}x_1\dots\mathrm{d}x_{l+1}\\
&\forall\, v\in\left(L^2(D)\right)^{\otimes l}\otimes H^1_{\Gamma_D}(D).
\end{aligned}}$$

Note that problem (3.2) is a particular case of (3.3) for $l=0$, since $\mathbb{E}\left[Y^{\otimes 0}\otimes u^{k-0}\right]=\mathbb{E}\left[u^k\right]$.

To summarize, the problem at hand has a *recursive structure*. To obtain the K-th order approximation of $\mathbb{E}\left[u\right]$ we perform the following algorithm:

```
for $k = 0,\dots,K$ do
    Compute $\mathbb{E}\left[Y^{\otimes k}\otimes u^0\right]$.
    for $l = k-1, k-2,\dots,0$ do
        Solve the boundary value problem (3.3) to obtain the $(l+1)$-point correlation function $\mathbb{E}\left[Y^{\otimes l}\otimes u^{k-l}\right]$.
    end for
    The result for $l = 0$ is the $k$-th order correction $\mathbb{E}\left[u^k\right]$ to the mean $\mathbb{E}\left[u\right]$.
end for
```

Table 1

K-th order approximation of the mean. The first column contains the input terms $\mathbb{E}\left[Y^{\otimes k}\otimes u^0\right]$ and the first row contains the k-th order corrections $\mathbb{E}\left[u^k\right]$, for $k=0,\dots,K$. To compute $\mathbb{E}\left[T^K u(Y,x)\right]$, we need all elements in the top left triangular part, that is, all elements in the k-th diagonals for $k=0,\dots,K$.

u^0	0	$\mathbb{E}\left[u^2\right]$	0	$\mathbb{E}\left[u^4\right]$	0
0	$\mathbb{E}\left[Y\otimes u^1\right]$	0	$\mathbb{E}\left[Y\otimes u^3\right]$	0	$\iddots$
$\mathbb{E}\left[Y^{\otimes 2}\right]\otimes u^0$	0	$\mathbb{E}\left[Y^{\otimes 2}\otimes u^2\right]$	0	$\iddots$	0
0	$\mathbb{E}\left[Y^{\otimes 3}\otimes u^1\right]$	0	$\iddots$	0	$\iddots$
$\mathbb{E}\left[Y^{\otimes 4}\right]\otimes u^0$	0	$\iddots$	0	$\iddots$	0

Table 1 illustrates the computational flow of our algorithm. Each non-zero correlation $\mathbb{E}\left[Y^{\otimes l}\otimes u^{k-l}\right]$, with $l<k$, can be obtained only if we have previously computed all the previous terms in the k-th diagonal. As a consequence, to derive the K-th order approximation $\mathbb{E}\left[T^K u(Y,x)\right]$, we need to compute all elements in the top left triangular part of the table. Notice that, since we assumed $\mathbb{E}\left[Y\right](x)=0$ w.l.o.g., all the $(2k+1)$-point correlations of Y vanish, and hence all odd diagonals are zero.

Remark 3.1.1. *The well-posedness of problem* (3.3) *is proved in [4] under the* *assumption* $\mathrm{Cov}_Y \in \mathcal{C}^{0,t}(\overline{D \times D})$ *for some* $0 < t \leq 1$. *Moreover, in [4] it is shown that further regularity assumptions on the domain and* u^0 *guarantee a regularity result for all* $\mathbb{E}\left[Y^{\otimes l} \otimes u^{k-l}\right]$ *in Hölder spaces with mixed regularity.*

4. Tensor Product Finite Element discretization.

To simplify the presentation, we suppose $g = 0$ from now on, that is, homogeneous Dirichlet boundary conditions are imposed on Γ_D in the stochastic Darcy problem. Let $\mathcal{T}_h$ be a regular partition of the domain $D \subseteq \mathbb{R}^d$ ($d = 1, 2, 3$) into intervals/triangles/tetrahedra, such that the union of all the elements in $\mathcal{T}_h$ is the closure of the domain and the intersection of any two elements is empty or is a common vertex or edge or face. Let us denote with h the discretization parameter, i.e. the maximum diameter of the elements in $\mathcal{T}_h$. To discretize the Hilbert spaces $H^1_{\Gamma_D}(D)$ and $L^2(D)$ we use piecewise linear and piecewise constant finite elements (FE) respectively:

$$V_h = \mathrm{span}\left\{\phi_n\right\}_{n=1}^{N_v} \subset H^1_{\Gamma_D}(D), \tag{4.1}$$

$$W_h = \mathrm{span}\left\{\psi_i\right\}_{i=1}^{N_e} \subset L^2(D), \tag{4.2}$$

where $\{\phi_n\}_{n=1}^{N_v}$ is the Lagrangian $\mathbb{P}_1$ basis, $\{\psi_i\}_{i=1}^{N_e}$ is the piecewise constant basis composed of the indicator functions for each element of the triangulation, N_v is the number of vertices excluding those on Γ_D, and N_e is the number of elements of the triangulation. Having the two bases $\{\phi_n\}$ and $\{\psi_i\}$, we can construct the *full tensor product* (FTP) basis for the tensor product space $\left(L^2(D)\right)^{\otimes l} \otimes H^1_{\Gamma_D}(D)$ for every integer $l \geq 1$:

$$(W_h)^{\otimes l} \otimes V_h = \mathrm{span}\left\{\psi_{i_1} \otimes \ldots \otimes \psi_{i_l} \otimes \phi_n,\ n = 1, \ldots, N_v,\ i_1, \ldots, i_l = 1, \ldots, N_e\right\}.$$

The resulting discretized representations of the terms $\mathbb{E}\left[Y^{\otimes l} \otimes u^{k-l}\right]$ and $\mathrm{Tr}_{|l+1:l+s+1}\mathbb{E}\left[Y^{\otimes(s+l)} \otimes \nabla u^{k-l-s}\right]$ are given by

$$\begin{aligned}
&\mathbb{E}\left[Y^{\otimes l} \otimes u^{k-l}\right](x_1, \ldots, x_{l+1}) \\
&\quad = \sum_{i_1,\ldots,i_l,n} \mathcal{C}_{l,k-l}(i_1, \ldots, i_l, n)\psi_{i_1}(x_1) \otimes \cdots \otimes \psi_{i_l}(x_l) \otimes \phi_n(x_{l+1}), \\
&\mathrm{Tr}_{|l+1:l+s+1}\mathbb{E}\left[Y^{\otimes(s+l)} \otimes \nabla u^{k-l-s}\right](x_1, \ldots, x_{l+1}) \\
&\quad = \sum_{i_1,\ldots,i_{s+l},n} \mathcal{C}_{s+l,k-l-s}(i_1, \ldots, i_{s+l}, n) \cdots \\
&\qquad\qquad \psi_{i_1}(x_1) \otimes \cdots \otimes \psi_{i_l}(x_l) \otimes \psi_{i_{l+1}}(x_{l+1}) \otimes \cdots \otimes \psi_{i_{s+l}}(x_{l+1}) \otimes \nabla\phi_n(x_{l+1}),
\end{aligned}$$

where $\mathcal{C}_{p,q} \in \mathbb{R}^{N_e \times \cdots \times N_e \times N_v}$ denotes the tensor of order $p+1$ that contains the (yet unknown) coefficients of the $(p+1)$-point correlation $\mathbb{E}\left[Y^{\otimes p} \otimes u^q\right]$ in the chosen tensor product FE basis.

We test the recursion relation (3.3) against $v(x_1, \cdots, x_{l+1}) = \psi_{j_1}(x_1) \otimes \cdots \otimes$

$\psi_{j_l}(x_l) \otimes \phi_m(x_{l+1})$ so that the left hand side in (3.3) becomes:

$$\begin{aligned}
&\sum_{i_1,\dots,i_l,n} \mathcal{C}_{l,k-l}(i_1,\dots,i_l,n) \\
&\quad \int_{D^{\times(l+1)}} (\psi_{i_1}(x_1)\otimes\cdots\otimes\psi_{i_l}(x_l)\otimes\nabla\phi_n(x_{l+1}))\cdot(\psi_{j_1}(x_1)\otimes\cdots\otimes\psi_{j_l}(x_l)\otimes\nabla\phi_m(x_{l+1})) \\
&= \sum_{i_1,\dots,i_l,n} \mathcal{C}_{l,k-l}(i_1,\dots,i_l,n) \\
&\quad \left(\int_D \psi_{i_1}(x_1)\psi_{j_1}(x_1)\mathrm{d}x_1\right)\cdots\left(\int_D \psi_{i_l}(x_l)\psi_{j_l}(x_l)\mathrm{d}x_l\right)\left(\int_D \nabla\phi_n(x_{l+1})\cdot\nabla\phi_m(x_{l+1})\mathrm{d}x_{l+1}\right) \\
&= \sum_{i_1,\dots,i_l,n} \mathcal{C}_{l,k-l}(i_1,\dots,i_l,n)M(i_1,j_1)\cdots M(i_l,j_l)A(n,m),
\end{aligned}$$

where A is the stiffness matrix of the $\mathbb{P}_1$ basis and M is the mass matrix of the $\mathbb{P}_0$ basis. In a similar way, we write each term

$$I = \int_{D^{\times(l+1)}} \mathrm{Tr}_{|_{l+1:l+s+1}} \mathbb{E}\left[Y^{\otimes(s+l)}\otimes\nabla u^{k-l-s}\right]\cdot\left(\mathrm{Id}^{\otimes l}\otimes\nabla\right) v \,\mathrm{d}x_1\dots\mathrm{d}x_{l+1}$$

in the right hand side of (3.3) as

$$I = \sum_{i_1,\dots,i_{s+l},n} \mathcal{C}_{s+l,k-l-s}(i_1,\dots,i_{s+l},n)\mathcal{B}^s(i_{l+1},\dots,i_{s+l},n,m)M(i_1,j_1)\cdots M(i_l,j_l),$$

where $\mathcal{B}^s \in \mathbb{R}^{N_e\times\dots\times N_e\times N_v\times N_v}$ is the tensor of order $s+2$ defined as

$$\mathcal{B}^s(i_{l+1},\dots,i_{s+l},n,m) := \int_D \psi_{i_{l+1}}(x)\dots\psi_{i_{l+s}}(x)\;\nabla\phi_n(x)\cdot\nabla\phi_m(x)\;\mathrm{d}x. \tag{4.3}$$

Since we have discretized the space $L^2(D)$ with piecewise constant elements, we have

$$\begin{aligned}
\mathcal{B}^s(i_{l+1},\dots,i_{l+s},n,m) &= \delta_{i_{l+s-1},i_{l+s}}\mathcal{B}^{s-1}(i_{l+1},\dots,i_{l+s-1},n,m) \\
&= \cdots = \delta_{i_{l+1},\dots,i_{l+s}}\mathcal{B}^1(i_{l+1},n,m),
\end{aligned} \tag{4.4}$$

where $\delta_{i_{l+1},\dots,i_{l+s}} = \begin{cases} 1, & \text{if } i_{l+1}=\dots=i_{l+s} \\ 0, & \text{otherwise} \end{cases}$. Hence, $\mathcal{B}^s$ is a highly sparse tensor that represents the discrete analogue of the trace operator $\mathrm{Tr}_{|_{l+1:l+s+1}}$. Thanks to relation (4.4), there is no need to explicitly compute $\mathcal{B}^s$ for $s \geq 2$, and it is sufficient to compute the tensor $\mathcal{B}^1$, representing the collection of local stiffness matrices.

The following definition represents the discrete analogue of Definition 3.1.

Definition 4.1. *Let $\mathcal{Y} \in \mathbb{R}^{n_1\times\dots\times n_p\times\dots\times n_q\times\dots\times n_r}$ and $\mathcal{X} \in \mathbb{R}^{n_p\times\dots\times n_q\times h}$ be tensors of order r and $q-p+2$, respectively, with $1 \leq p \leq q \leq r$. We define the tensor of order $p-q+r$*

$$\mathcal{Z} := \mathcal{X} \times_{p:q} \mathcal{Y} \in \mathbb{R}^{n_1\times\dots\times n_{p-1}\times h\times n_{q+1}\times\dots\times n_r} \tag{4.5}$$

as the contraction of the first $q-p+1$ indices of $\mathcal{X}$ with the indices of $\mathcal{Y}$ from position p until q:

$$\begin{aligned}
&\mathcal{Z}(k_1,\dots,k_{p-1},j,k_{q+1},\dots,k_r) \\
&\quad = \sum_{k_p=1}^{n_p}\cdots\sum_{k_q=1}^{n_q} \mathcal{X}(k_p,\dots,k_q,j)\mathcal{Y}(k_1,\dots,k_p,\dots,k_q,\dots,k_r).
\end{aligned} \tag{4.6}$$

In the case $p = q$ we write $\times_p$ instead of $\times_{p:p}$.

Remark 4.1.1. *If $p = q$ in Definition 4.1 then $\mathcal{X}(k_p, j)$ is a matrix and $\mathcal{Z} := \mathcal{X} \times_{p:p} \mathcal{Y} = \mathcal{X} \times_p \mathcal{Y}$ coincides with the usual p-mode product of a tensor with a matrix [22].*

Remark 4.1.2. *Given a tensor $\mathcal{Y}$ of order $p + s$, let us set $\mathcal{X} = \mathcal{B}^s$ and define $\mathcal{Z} := \mathcal{B}^s \times_{p:p+s} \mathcal{Y}$. Then, using* (4.4)*, we can write the elements of $\mathcal{Z}$ as*

$$\begin{aligned}
&\mathcal{Z}(i_1, \dots, i_{p-1}, m) \\
&\quad = (\mathcal{B}^s \times_{p:p+s} \mathcal{Y})\,(i_1, \dots, i_{p-1}, m) \\
&\quad = \sum_{i_p, \dots, i_{s+p-1}, n} \mathcal{B}^s(i_p, \dots, i_{s+p-1}, n, m)\mathcal{Y}(i_1, \dots, i_{p-1}, i_p, \dots, i_{p+s-1}, n) \\
&\quad = \sum_{i_p, \dots, i_{s+p-1}, n} \delta_{i_p, \dots, i_{p+s-1}} \mathcal{B}^1(i_p, n, m)\mathcal{Y}(i_1, \dots, i_{p-1}, i_p, \dots, i_{p+s-1}, n) \\
&\quad = \sum_i \sum_n \mathcal{B}^1(i, n, m)\mathcal{Y}(i_1, \dots, i_{p-1}, \underbrace{i, \dots, i}_{s \text{ times}}, n). \qquad (4.7)
\end{aligned}$$

We have already observed that to compute $\mathcal{Z} := \mathcal{B}^s \times_{p:p+s} \mathcal{Y}$ it is sufficient to compute the tensor $\mathcal{B}^1$. However, even the storage of the full tensor $\mathcal{B}^1$, which has size $N_e \times N_v \times N_v$, is impractical for 2D and 3D meshes. In our algorithm, we do not compute the full tensor $\mathcal{B}^1$, but we rather perform a loop over the elements of the mesh $i = 1, \dots, N_e$ and at each step of the loop the local (element-wise) stiffness matrix $B_i := \mathcal{B}^1(i, :\,.\,:)$ is constructed and used to perform the operation in (4.7)*. See Algorithm 2 below for more details. Notice that all local stiffness matrices can also be precomputed and stored during the simulation.*

Using Definition 4.1 and observing that $M(i, j) = |T_j|\delta_{i,j}$, where $|T_j|$ denotes the area of the j-th element, we can write the FE formulation of problem (3.3) in a compact way:

$$\boxed{A \times_{l+1} \mathcal{C}_{l,k-l} = -\sum_{s=1}^{k-l} \begin{pmatrix} k-l \\ s \end{pmatrix} \mathcal{B}^s \times_{l+1:l+s+1} \mathcal{C}_{s+l,k-l-s}.} \qquad (4.8)$$

Remark 4.1.3. *As an alternative to the full tensor product discretization of the tensor product space $\left(L^2(D)\right)^{\otimes l} \otimes H^1_{\Gamma_D}(D)$, we mention the possibility of using sparse tensor product spaces, which considerably limit the number of degrees of freedom while keeping almost the same accuracy. For more details, we refer, e.g., to [6, 29]. In this paper, we follow the full tensor product technique and then represent the obtained tensors in low-rank format. Notice that a sparse grid representation can be seen as a particular case of low-rank representation with a fixed choice of bases [19].*

5. Tensor Train format.

The FE discretization of the k-th order problem (4.8) involves high-order tensors. As the number of entries in a tensor grows exponentially with the order, only low-order tensors can be stored explicitly. This phenomenon is known as the *curse of dimensionality.* For higher-order tensors it is necessary to use low-rank formats.

Classical low-rank formats for tensors include the CP (Canonical decomposition/Parallel factors) format [22, 7, 21] and the Tucker format [22, 33]. Note that the latter format is restricted to tensors of moderate order. We refer to the Tensor Toolbox [1] for a Matlab implementation of both formats.

Recently, two other approaches have been proposed to address high-dimensional problems in scientific computing: the Hierarchical Tucker (HT) format [16, 20] and the Tensor Train (TT) format [26]; see [24, 25] for MATLAB implementations. Both formats are based on the singular value decomposition (SVD) and require the storage of a collection of third-order tensors instead of the original higher-order tensor. The simpler structure of the TT-format, which can be seen as a particular case of the HT-format, makes it easier to handle compared to the HT-format. For this reason, we have based our developments on the TT-format.

DEFINITION 5.1. *A tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$ of order d is in* TT-format *if it is represented as*

$$\mathcal{X}(i_1, \dots, i_d) = \sum_{\alpha_1=1}^{r_1} \dots \sum_{\alpha_{d-1}=1}^{r_{d-1}} G_1(i_1, \alpha_1) G_2(\alpha_1, i_2, \alpha_2) \dots G_d(\alpha_{d-1}, i_d). \tag{5.1}$$

The third-order tensors $G_j \in \mathbb{R}^{r_{j-1} \times n_j \times r_j}$ for $j = 1, \dots, d$, with $r_0 = r_d = 1$, are called the cores *of the TT-format.*

The tuple $(r_1, \dots, r_{d-1})$ critically determines the storage complexity of the TT-format, which is given by $O\big((d-2)nr^2 + 2rn\big)$ where $n = \max\{n_1, \dots, n_d\}$, $r = \max\{r_1, \dots, r_d\}$. The smallest possible tuple $(r_1, \dots, r_{d-1})$ admitting a representation (5.1) for a given tensor $\mathcal{X}$ is called the *TT-rank* of $\mathcal{X}$. The TT-rank corresponds to the ranks of certain matricizations (flattenings) of $\mathcal{X}$. One of the most important features of the TT-format is that the compression of a given tensor to a tensor with prescribed (low) TT-rank can be performed by means of singular value decompositions within quasi-optimal accuracy [26]. The same holds for recompressing a tensor in TT-format to lower TT-rank. Moreover, many linear algebra operations (like addition, scalar product, μ-th mode product, etc.) can be cast efficiently for such tensors in terms of operations on the cores only. All these operations are implemented in the Matlab TT-Toolbox [25].

To emphasize that a tensor $\mathcal{X}$ is given in terms of the TT-format (5.1), we will write $\mathcal{X}^{\text{TT}}$ instead of $\mathcal{X}$.

6. The recursive TT-algorithm.

In this section we describe our algorithm for solving the recursive FE problem (4.8) in TT-format. Let us recall that the inputs of the recursion are the k-point correlations $\mathbb{E}\left[Y^{\otimes k}\right]$, whereas the outputs consist of the corrections $\mathbb{E}\left[u^k\right]$ of increasing order.

6.1. Approximation of the inputs in TT-format.

Before starting the recursion, we need to represent the k-point correlations $\mathbb{E}\left[Y^{\otimes k}\right]$ in an effective way. For this purpose, we follow a technique proposed in [23] and use a tensor $\mathcal{C}_k^{\text{TT}}$ in TT-format to represent $\mathbb{E}\left[Y^{\otimes k}\right]$ approximately. For completeness, we recall the main steps of the construction from [23].

Step 1 Computation of the truncated KL-expansion for the field $Y(\omega, x)$:

$$Y_N(\omega, x) = \sigma \sum_{j=1}^{N} \sqrt{\lambda_j}\ \eta_j(x)\ \xi_j(\omega), \tag{6.1}$$

where $\{\lambda_j\}_{j=1}^N$ are the non-negative eigenvalues in decreasing order and $\{\eta_j\}_{j=1}^N$ are the corresponding eigenvectors of the covariance function. At this step, N is chosen sufficiently large to lead to a negligible truncation error.

Step 2 Given an even order k and the set $\{\lambda_j\}_{j=1}^N$, the correlation $\mathbb{E}\left[Y_N^{\otimes k}\right]$ is given by

$$\begin{aligned}\mathbb{E}\left[Y_N^{\otimes k}\right](x_1,\dots,x_k) &= \sigma^k \sum_{i_1=1}^{N}\dots\sum_{i_k=1}^{N} \mathbb{E}\left[\prod_{\mu=1}^{k}\sqrt{\lambda_{i_\mu}}\xi_{i_\mu}(\omega)\right]\bigotimes_{\mu=1}^{k}\eta_{i_\mu}\\ &= \sigma^k \sum_{i_1=1}^{N}\dots\sum_{i_k=1}^{N} \underbrace{\prod_{l=1}^{N}\lambda_l^{m_{\mathbf{i}}(l)/2}\mathbb{E}\left[\xi_l(\omega)^{m_{\mathbf{i}}(l)}\right]}_{=:\mathcal{G}_k(i_1,\dots,i_k)}\bigotimes_{\mu=1}^{k}\eta_{i_\mu},\end{aligned}$$

where $m_{\mathbf{i}}(l) := \#\{n \ : \ i_n = l\}$ is the multiplicity of the integer l in the multi-index $\mathbf{i} = (i_1,\dots,i_k)$. Even when exploiting its super-symmetry, the explicit computation and representation of the entries $\mathcal{G}_k(i_1,\dots,i_k)$ in the core tensor $\mathcal{G}_k$ becomes way too expensive already for moderately large values of k. To avoid this, we approximate $\mathcal{G}_k$ by a tensor $\mathcal{G}_k^{TT}$ in TT-format such that

$$\|\mathcal{G}_k - \mathcal{G}_k^{TT}\|_F \leq \mathsf{tol} \tag{6.2}$$

for a prescribed tolerance $\mathsf{tol} > 0$. Note that $\|\cdot\|_F$ denotes the usual Frobenius norm of a tensor. Since the basis $\{\eta_j\}_j$ is orthonormal in $L^2(D)$, we have $\left\|\mathbb{E}\left[Y_N^{\otimes k}\right]\right\|_{(L^2(D))^{\otimes k}} = \sigma^k \|\mathcal{G}_k\|_F$. The function **constr_tt** described in [23, 31] that attains (6.2) consists of successively pruning those columns from the matricizations of $\mathcal{G}_k$ that only have a negligible impact on the overall approximation quality. Exploiting the specific structure of the entries of $\mathcal{G}_k$, this procedure does not need to form $\mathcal{G}_k$ explicitly, which enables it to handle relatively large values of k, such as $k = 10$ or $k = 20$.

Step 3 Using the function **tt_round** from the TT-toolbox, the tensor $\mathcal{G}_k^{TT}$ is compressed further to lower TT-ranks while maintaining (6.2), possibly with a different tolerance. Multiplying by σ^k, we finally obtain an approximation of $\mathbb{E}\left[Y_N^{\otimes k}\right]$ in TT-format represented in the KL-basis. This approximation will be denoted by $\mathcal{C}_k^{\mathrm{TT}}$.

Remark 6.0.1. *For convenience we store $\mathcal{C}_k^{TT}$, representing the correlation $\mathbb{E}\left[Y_N^{\otimes k}\right]$ in the KL-basis. We then convert each core separately into the FE basis when needed during the computations.*

Algorithm 1 summarizes Steps 2 and 3 above; it can be applied to a d-dimensional domain D, with $d \geq 1$. Observe that the cost of computing $\mathcal{C}_k^{\mathrm{TT}}$ and its storage is independent of the number of degrees of freedom N_h in the FE discretization and depends only on the decay of the eigenvalues $\{\lambda_j\}$ or, equivalently, the truncation level $\tilde{N} \leq N$ needed to achieve a prescribed accuracy by the KL expansion.

Algorithm 1 Function **compute_moment_Y**: Computes the k-point correlation of a centered Gaussian random field, with k even.

```
Require: Order of the correlation $k$, standard deviation $\sigma$, eigenvector matrix $\eta = (\eta_1,\dots,\eta_N)$, eigenvalues $\lambda = (\lambda_1,\dots,\lambda_N)$ and tolerances $\mathsf{tol}_1$, $\mathsf{tol}_2$
Ensure: Tensor $\mathcal{C}_k^{\mathrm{TT}}$ in TT-format approximating $\mathbb{E}\left[Y_N^{\otimes k}\right]$
  $\mathcal{G}_k^{TT} = \mathbf{constr\_tt}(k, \lambda, \mathsf{tol}_1)$
  $\mathcal{C}_k^{\mathrm{TT}} = \mathbf{tt\_round}(\mathcal{G}_k^{TT}, \mathsf{tol}_2)$
  $\mathcal{C}_k^{\mathrm{TT}} \leftarrow \sigma^k \mathcal{C}_k^{\mathrm{TT}}$
```

Remark 6.0.2. *Note that the tensor $\mathbb{E}\left[Y_N^{\otimes k}\right]$ is supersymmetric, i.e., it is invariant under any permutation of its indices. The compressions resulting in $\mathcal{C}_k^{\mathrm{TT}}$ do not fully preserve these symmetries, but we still have*

$$\mathcal{C}_k^{\mathrm{TT}}(i_1,\dots,i_{k/2},i_{k/2+1},\dots,i_k) = \mathcal{C}_k^{\mathrm{TT}}(i_k,\dots,i_{k/2+1},i_{k/2},\dots,i_1).$$

6.2. Computation of the mean corrections in TT-format. Let K be even and fixed. The computation of the K-th order correction requires to recursively solve the linear systems (4.8) for $k = 2,\dots,K$ and $l = k-1, k-2,\dots,0$; see also Table 1. Starting from the tensor $\mathcal{C}_k^{\mathrm{TT}}$ computed above (which corresponds to $l = k$), we now aim at carrying out these recursions in the TT-format. The main ingredients needed to attain this goal are the implementation of contractions of the form $\mathcal{B}^s \times_{p:p+s} \mathcal{X}^{\mathrm{TT}}$, with $\mathcal{X}^{\mathrm{TT}}$ being a TT-tensor of order $p+s$, and the solution of linear systems of the form $A \times_p \mathcal{X}^{\mathrm{TT}} = \mathcal{Y}^{\mathrm{TT}}$, with A being the FE stiffness matrix and $\mathcal{Y}^{\mathrm{TT}}$ being a tensor of order p in TT-format.

6.2.1. Contraction $\mathcal{B}^s \times_{p:p+s} \mathcal{X}^{\mathrm{TT}}$ in TT-format. We have implemented the function **contraction** (see Algorithm 2) which, for a given $(p+s)$-th order tensor $\mathcal{X}^{\mathrm{TT}}$ in TT-format, with $p,s \geq 1$, returns the p-th order tensor $\mathcal{Z}^{\mathrm{TT}} := \mathcal{B}^s \times_{p:p+s} \mathcal{X}^{\mathrm{TT}}$ element-wise defined by (4.7).

The first step of the algorithm consists of extracting the cores from the TT-format of $\mathcal{X}^{\mathrm{TT}}$. This operation is realized by the sub-routine **extract_cores**, which, given the TT-tensor $\mathcal{X}^{\mathrm{TT}}$, returns a cell array $\mathcal{X}_{\mathrm{cores}}$ containing the cores of $\mathcal{X}^{\mathrm{TT}}$. We then initialize a cell array $\mathcal{Z}_{\mathrm{cores}}$ of length p. Since the contraction (4.7) does not affect the first $p-1$ modes of the tensor, the corresponding cores of $\mathcal{X}^{\mathrm{TT}}$ and $\mathcal{Z}^{\mathrm{TT}}$ are identical. To compute the p-th core of $\mathcal{Z}^{\mathrm{TT}}$ we perform a loop over the elements of the triangulation. In each step of the loop, the local stiffness matrix $\mathcal{B}^1(i,:,:)$ is constructed and multiplied with the $(p+s)$-th core $\mathcal{X}_{\mathrm{cores}}\{p+s\}$. The result is then recursively multiplied with the j-th core of $\mathcal{X}^{\mathrm{TT}}$ for $j = p,\dots,p+s-1$. Note that the matrix $\mathcal{B}^1(i,:,:)$ is highly sparse, since it coincides with the stiffness matrix corresponding to the i-th element of the mesh. Finally, the tensor $\mathcal{Z}^{\mathrm{TT}}$ is assembled in TT-format from its cores by calling the function **tt_tensor** from the TT-Toolbox.

Remark 6.0.3. *Algorithm 2 assumes that the tensor $\mathcal{X}^{\mathrm{TT}}$ is represented in the FE basis. On the other hand, if the tensor $\mathcal{X}^{\mathrm{TT}}$ is represented in the KL basis, each core $\mathcal{X}_{\mathrm{cores}}\{j\}(:,i,:)$ in the inner loop has to be recast in the FE basis before performing the next operation. Note that the p-th core of $\mathcal{Z}^{\mathrm{TT}}$ will always be in FE basis, whereas the first $p-1$ cores will remain expanded in the KL basis.*

6.2.2. Solution of the linear system $A \times_p \mathcal{X}^{\mathrm{TT}} = \mathcal{Y}^{\mathrm{TT}}$ in TT-format. Given a p-th order tensor $\mathcal{Y}^{\mathrm{TT}}$ in TT-format and the $N_v \times N_v$ stiffness matrix A, the function **solve_linear_system** described in Algorithm 3 returns the solution of the linear system $A \times_p \mathcal{X}^{\mathrm{TT}} = \mathcal{Y}^{\mathrm{TT}}$ in TT-format. The p-th core (the last one) of $\mathcal{X}^{\mathrm{TT}}$ is obtained from the p-th core of $\mathcal{Y}^{\mathrm{TT}}$ by applying A^{-1}. This corresponds to solving $r_{p-1} \cdot r_p$ systems with the same matrix A. All other cores of $\mathcal{X}^{\mathrm{TT}}$ and $\mathcal{Y}^{\mathrm{TT}}$ are identical. In practice we have used the backslash Matlab operator to solve these linear systems.

6.2.3. Recursive first moment problem in TT-format. Using the ingredients discussed above, Algorithm 4 computes the corrections $\mathbb{E}\left[u^k\right]$ for $k = 0,\dots,K$ in TT-format.

After solving the 0-th order approximation problem, we compute the truncated KL-expansion of the Gaussian random field Y. We then loop over the order k of

Algorithm 2 Function **contraction**: Performs the contraction $\mathcal{B}^s \times_{p:p+s} \mathcal{X}^{\text{TT}}$ in TT-format.

```
Require: Tensor $\mathcal{X}^{\text{TT}}$ of order $p+s$ in TT-format, and positive integers $p, s$ with
   $p, s \geq 1$.
Ensure: Tensor $\mathcal{Z}^{\text{TT}} = \mathcal{B}^s \times_{p:p+s} \mathcal{X}^{\text{TT}}$ in TT-format, as defined in (4.7).
  Set $\mathcal{X}_{\text{cores}} =$ extract_cores($\mathcal{X}^{\text{TT}}$)
  Initialize an empty cell array $\mathcal{Z}_{\text{cores}}$ of length $p$
  Set $\mathcal{Z}_{\text{cores}}\{j\} = \mathcal{X}_{\text{cores}}\{j\}$ for $j = 1, \dots, p-1$
  for $i = 1, \dots, N_e$ do
      Compute the matrix $B_i = \mathcal{B}^1(i,:,:)$
      Set $\text{temp}_i(\alpha_{p+s-1}, m, \alpha_{p+s}) = \sum_n \mathcal{X}_{\text{cores}}\{p+s\}(\alpha_{p+s-1}, n, \alpha_{p+s}) B_i(n,m)$
      for $j = p+s-1 : -1 : p$ do
          Extract the matrix $X_i = \mathcal{X}_{\text{cores}}\{j\}(:,i,:)$
          $\text{temp}_i(\alpha_{j-1}, m, \alpha_{p+s}) = \sum_{\alpha_j} \text{temp}_i(\alpha_j, m, \alpha_{p+s}) X_i(\alpha_{j-1}, \alpha_j)$
      end for
      $\mathcal{Z}_{\text{cores}}\{p\} \leftarrow \mathcal{Z}_{\text{cores}}\{p\} + \text{temp}_i$
  end for
  Set $\mathcal{Z}^{\text{TT}} =$ tt_tensor($\mathcal{Z}_{\text{cores}}$)
```

Algorithm 3 Function **solve_linear_system**: Solves $A \times_p \mathcal{X}^{\text{TT}} = \mathcal{Y}^{\text{TT}}$.

```
Require: Matrix $A \in \mathbb{R}^{N_v \times N_v}$ and tensor $\mathcal{Y}^{\text{TT}} \in \mathbb{R}^{N_e \times \dots \times N_e \times N_v}$ of order $p$ in TT-
   format.
Ensure: Solution $\mathcal{X}^{\text{TT}} \in \mathbb{R}^{N_e \times \dots \times N_e \times N_v}$ of $A \times_p \mathcal{X}^{\text{TT}} = \mathcal{Y}^{\text{TT}}$ in TT-format.
  $\mathcal{Y}_{\text{cores}} =$ extract_cores($\mathcal{Y}^{\text{TT}}$)
  Set $\mathcal{X}_{\text{cores}} = \mathcal{Y}_{cores}$
  Set $\mathcal{X}_{\text{cores}}\{p\} = A \backslash \mathcal{Y}_{\text{cores}}\{p\}$
  Set $\mathcal{X}^{\text{TT}} =$ tt_tensor($\mathcal{X}_{\text{cores}}$)
```

the correction. In each step we first compute the tensors $\mathcal{C}_k^{\text{TT}}$ and $\mathcal{C}_{k,0}^{\text{TT}}$ in TT-format, representing the k-point correlation functions $\mathbb{E}\left[Y^{\otimes k}\right]$ and $\mathbb{E}\left[Y^{\otimes k} \otimes u^0\right]$, respectively. For the latter, we use the function **kron** from the TT-toolbox for performing the Kronecker product in the TT-format. In a nested loop over the index $l = k-1, k-2, \dots, 1, 0$, we compute the right-hand side of (4.8) using the function **contraction** described in Algorithm 2), and solve the linear system (4.8) using the function **solve_linear_system** described in Algorithm 3. Finally, we compute the K-th order correction $\mathbb{E}\left[T^K u\right]$ by summing up all corrections.

Remark 6.0.4. *We remark that the assembly of the right hand side of* (4.8) *involves the sum of tensors in TT-format. Generically, the TT-ranks of the resulting TT-tensor are given by the sum of the TT-ranks of all the terms in the sum. Recompression needs to be performed to avoid this growth of the TT-ranks. More specifically, the application of* ***tt_round*** *after each addition, with a tolerance* $\mathsf{tol}_3 \leq \mathsf{tol}_2$, *can be expected to lead to much smaller ranks, especially in the cores l and $l+1$. On the other hand, performing this procedure requires considerable amounts of memory.*

For the particular situation at hand, it turns out that the summation in (4.8) *can be performed in a simpler way because all terms have the first l cores in common. Hence, these cores will stay the same in the (exact) sum and only the* $(l+1)$*-th core needs to be computed, its rank being equal to the l-th rank r_l of $\mathcal{C}_{k,0}^{\text{TT}}$. To benefit from* *this property, we do not perform compression during the summation as it would modify also the first l cores.*

Algorithm 4 Implementation of the recursion in Table 1.

```
Require: Order of the approximation $K$, standard deviation $\sigma$, covariance function
  $\mathrm{Cov}_Y$ of $\frac{1}{\sigma}Y$, tolerances $\mathsf{tol}_{\mathsf{KL}}$, $\mathsf{tol}_1$, $\mathsf{tol}_2$, $\mathsf{tol}_3$ load function $f$
Ensure: $K$-th order approximation of $\mathbb{E}[u]$ in TT-format, that is $\sum_{k=0}^{K} \frac{1}{k!}\mathcal{C}^{\mathrm{TT}}_{0,k}$
  Solve the deterministic problem for $\mathcal{C}_{0,0}$
  Given $\mathrm{Cov}_Y$, compute the truncated KL-expansion (6.1) of $Y$ up to a prescribed
  tolerance $\mathsf{tol}_{\mathsf{KL}}$.
  for $k = 2 : 2 : K$ do
      Construct $\mathcal{C}^{\mathrm{TT}}_{k} = \mathbf{compute\_moment\_Y}(k, \sigma, \eta, \lambda, \mathsf{tol}_1, \mathsf{tol}_2)$
      Compute $\mathcal{C}^{\mathrm{TT}}_{k,0} = \mathbf{kron}(\mathcal{C}^{\mathrm{TT}}_{k}, \mathcal{C}_{0,0})$
      for $l = k-1, \dots, 0$ do
          Initialize $\mathcal{Y}^{\mathrm{TT}} = 0$
          for $s = 1, \dots, k-l$ do
              Update $\mathcal{Y}^{\mathrm{TT}} \leftarrow \mathcal{Y}^{\mathrm{TT}} - \binom{k-l}{s} \mathcal{B}^{s} \times_{l+1:s+1} \mathcal{C}^{\mathrm{TT}}_{s+l,k-l-s}$ using the
              function contraction
              Recompress $\mathcal{Y}^{\mathrm{TT}} \leftarrow \mathbf{tt\_round}(\mathcal{Y}^{\mathrm{TT}}, \mathsf{tol}_3)$
          end for
          Set $\mathcal{C}^{\mathrm{TT}}_{l,k-l} = \mathbf{solve\_linear\_system}(A, \mathcal{Y}^{\mathrm{TT}})$
      end for
  end for
  Return $\sum_{k=0:2:K} \frac{\mathcal{C}^{\mathrm{TT}}_{0,k}}{k!}$
```

7. Storage requirement and complexity of the TT-algorithm. The storage complexity of a tensor of order k in TT-format highly depends on its TT-rank $(r_1, \dots, r_{k-1})$. Here we numerically study the storage complexity of the input data of our algorithm, that is the k-point correlations of Y in TT-format for $k = 0, \dots, K$, with k even. We aim also at understanding how this complexity spreads throughout the recursive problem described in Table 1. All our computations are performed in the one dimensional case, $D = [0, 1]$.

As described in Section 6.1, to compute the k-point correlation of Y in TT-format, $\mathcal{C}^{\mathrm{TT}}_{k}$, for $k \le K$ even, we first have to perform the truncated KL-expansion (6.1) of Y with a prescribed accuracy $\mathsf{tol}_{\mathsf{KL}}$. Throughout this section we take $\mathsf{tol}_{\mathsf{KL}} = 10^{-16}$, so that – up to machine precision – the *complete KL-expansion* corresponding to the piecewise constant discretization Y_h of the random field is considered.

Let N be the number of random variables parametrizing the field Y_h. Then $N \le N_e$, where N_e is the number of elements of the partition $\mathcal{T}_h$. In [23] it has been shown that if $\mathcal{C}^{\mathrm{TT}}_{k}$ were to be computed exactly, its TT-rank would satisfy

$$r_p = \binom{N+p-1}{p} \tag{7.1}$$

for $p = 1, \dots, k/2$. Because of the symmetry in the construction of $\mathcal{C}^{\mathrm{TT}}_{k}$ (see Remark 6.0.2), $r_{k-p} = r_p$ for $p = 1, \dots, k/2$. Clearly, the storage of the exact TT-format $\mathcal{C}^{\mathrm{TT}}_{k}$ becomes costly for moderately large k. Using Algorithm 1, we construct an approximation of this TT-tensor, which we still denote with $\mathcal{C}^{\mathrm{TT}}_{k}$. Its TT-rank is bounded by (7.1), but it is often observed to be significantly lower.

As an example, let us consider the domain $D = [0,1]$ discretized with $N_h = 100$ ($N_h = N_e$) subintervals of length $h = 1/N_h$, and the squared exponential covariance function

$$\mathrm{Cov}_Y(x_1, x_2) = \mathrm{e}^{-\frac{\|x_1 - x_2\|^2}{L^2}}, \quad (x_1, x_2) \in D \times D \tag{7.2}$$

with correlation length $L = 0.2$. The field is parametrized by $N = 26$ random variables. In Figure 1 we compare the upper bound in (7.1) (black dashed line) with the TT-ranks of $\mathcal{C}_6^{\mathrm{TT}}$ computed using Algorithm 1 and imposing different tolerances $\mathsf{tol}_1 = \mathsf{tol}_2 = \mathsf{tol}$. Smaller tolerances lead to higher TT-ranks. In Figure 2 (left) the same type of plot is done for $k = 2, 4, 6$.

Figure 2 (right) is obtained with $N_h = 100$ and the exponential covariance function

$$\mathrm{Cov}_Y(x_1, x_2) = \mathrm{e}^{-\frac{\|x_1 - x_2\|}{L}}, \quad (x_1, x_2) \in D \times D \tag{7.3}$$

with correlation length $L = 0.2$. $N = 100$ random variables are considered. In the exponential case, the TT-ranks are observed to grow faster than in the Gaussian setting. For more examples and details we refer to [23].

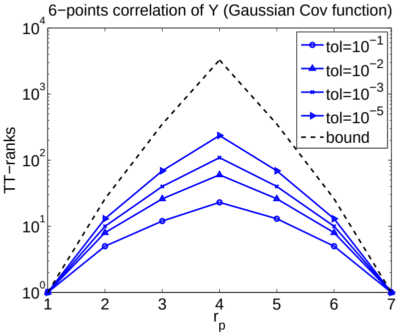

FIGURE 1. *Upper bound* (7.1) *(black dashed line) for the TT-ranks compared with the TT-ranks of the approximation* $\mathcal{C}_6^{\mathrm{TT}}$ *computed for different tolerances* tol.

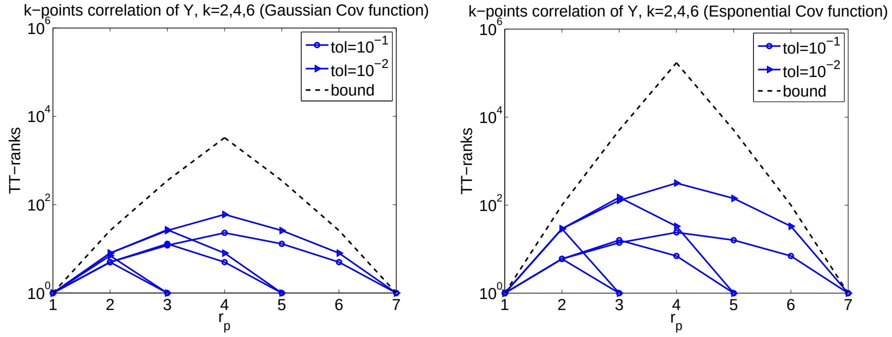

FIGURE 2. *Upper bound* (7.1) *(black dashed line) for the TT-ranks compared with the TT-ranks of the approximation* $\mathcal{C}_k^{\mathrm{TT}}$ *computed for different tolerances* tol. *Left plot:* $k = 2, 4, 6$ *and* Cov_Y *Gaussian. Right plot:* $k = 2, 4, 6$ *and* Cov_Y *exponential.*

We perform a numerical test to study the storage complexity of the correlations involved in the K-th order problem, that is, the entries in Table 1. As already emphasized in Remark 6.0.4, the most delicate operation in solving (4.8) is the sum of the TT-tensors needed in the construction of the right hand side. We have implemented and tested both ways described in Remark 6.0.4 for performing this operation:

1. We exploit that all terms in the sum have the first l cores in common and only the $(l+1)$-th core needs to be computed. No recompression is performed. Figure 3 displays the resulting TT-ranks of the correlations needed to compute the 6-th order correction term $\mathcal{C}^{\mathrm{TT}}_{0,6}$. We have used tolerances $\mathsf{tol}_1 = \mathsf{tol}_2 = 10^{-10}$ in the function **compute_moment_Y**. Note that the storage requirement is decreasing along each diagonal of Table 1, and the biggest effort is actually required for storing the input term $\mathcal{C}^{\mathrm{TT}}_{6,0}$.
2. We use the addition of TT-tensors implemented in the Matlab TT-Toolbox [25]. After each addition, recompression using **tt_round** is performed (see Algorithm 4). Figure 7 displays quantities analogous to Figure 3, with recompression tolerances $\mathsf{tol}_3 = 10^{-13}$ (left plot) and $\mathsf{tol}_3 = 10^{-6}$ (right plot). Clearly, this procedure leads to smaller ranks, at the expense of higher memory requirements and the computational cost needed for the recompressions.

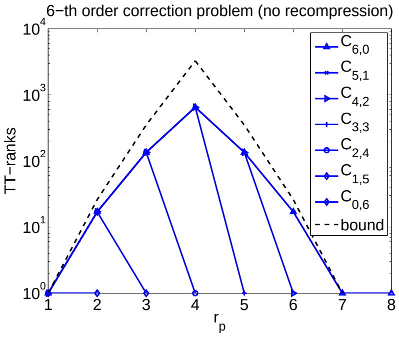

FIGURE 3. *Semilogarithmic plot of the TT-ranks of the correlations needed to solve the* 6*-th order problem, with tolerances* $\mathsf{tol}_{\mathsf{KL}} = 10^{-16}$ *and* $\mathsf{tol}_1 = \mathsf{tol}_2 = 10^{-10}$. *No compression is performed in summing the TT-tensors in the right hand side of* (4.8).

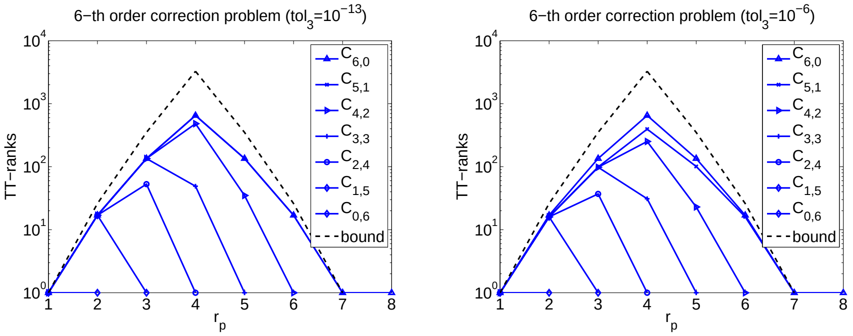

FIGURE 4. *Semilogarithmic plot of the TT-ranks of the correlations needed to solve the* 6*-th order problem, with tolerances* $\mathsf{tol}_{\mathsf{KL}} = 10^{-16}$ *and* $\mathsf{tol}_1 = \mathsf{tol}_2 = 10^{-10}$. *A recompression up to tolerances* $\mathsf{tol}_3 = 10^{-13}$ *(left) and* $\mathsf{tol}_3 = 10^{-6}$ *(right) is performed is performed in summing the TT-tensors in the right hand side of* (4.8).

Remark 7.0.5. *Given a d-variate, 2π-periodic function f with mixed Sobolev regularity s, the results in [28] show that the storage complexity for achieving accuracy* tol *by representing f in HT-format satisfies worse rates than those obtained with sparse grids. Actually, sparse grids are expressly constructed to approximate multivariate functions with mixed Sobolev regularity. How sparse grid techniques compare with low-rank approximations in other settings is still an open question and a matter of research.*

We give, in what follows, some considerations on the complexity of our method. Let N be the number of random variables we take into account in the KL-expansion, so that the random field Y is parametrized by the Gaussian random vector $\mathbf{Y} = (Y_1 \dots, Y_N)$.

In the approach proposed here, the first moment problem is derived and solved in TT-format. To compute the K-th order approximation $\mathbb{E}\left[T^K u\right]$ it is necessary to compute $\mathbb{E}\left[u^k\right]$ for $k = 0, 2, \dots, K$, and to compute each k-th order correction $\mathbb{E}\left[u^k\right]$ it is necessary to derive $\mathbb{E}\left[Y^{\otimes l} \otimes u^{k-l}\right]$, $l = k-1, k-2, \dots, 0$. Let $(r_0, r_1, \dots, r_k, r_{k+1})$ be the TT-rank of $\mathbb{E}\left[Y^{\otimes k} \otimes u^0\right]$, with $r_0 = r_{k+1} = 1$. At the computational level, solving the PDE for $\mathbb{E}\left[Y^{\otimes l} \otimes u^{k-l}\right]$ requires to solve a number of linear systems not larger than r_l (depending on whether a **tt_round** up to tolerance tol_3 is used in Algorithm 4). Hence, the computation of $\mathbb{E}\left[T^K u\right]$ entails the solution of

$$M_1 := \sum_{k=2:2:K} \sum_{l=0}^{k-1} r_l + 1$$

linear systems. If no compression in the construction of $\mathcal{C}^{\mathrm{TT}}_{K,0}$ is used, M_1 can be bounded using (7.1) as

$$\begin{aligned} M_1 &= \sum_{k=2:2:K} \sum_{l=0}^{k-1} r_l + 1 = \sum_{k=2:2:K} \sum_{l=0}^{k-1} \binom{N+l-1}{l} + 1 \\ &= \sum_{k=2:2:K} \binom{N+k-1}{k-1} + 1 \leq \binom{N+K}{K-1}. \end{aligned}$$

Using recompression in the construction of $\mathcal{C}^{\mathrm{TT}}_{K,0}$, as implemented in Algorithm 1 (see also [23]), the TT-format offers the possibility to dramatically reduce the ranks and hence the computational cost. In Section 8.2 we will refer to this reduced computational cost as M_1'.

The most natural alternative to the approach considered here is to directly compute the Taylor polynomial, and use it to approximate $\mathbb{E}[u]$. Since the number of partial derivatives of order k of a function of N variables is $\binom{N+k-1}{k}$, the direct computation of $T^K u$ entails the solution of

$$M_2 = \sum_{k=0}^{K} \binom{N+k-1}{k} = \binom{N+K}{K}$$

linear systems. We note therefore that, even without recompression, $M_1 < M_2$. On the other hand, M_2 reduces since only even moments of Gaussian random variables are non-zero, and can be further improved taking into account anisotropic Taylor polynomials. How an optimally truncated Taylor expansion compares with our moment equations approach with compressed TT-format is still an open question.

8. Numerical tests. In this section we perform some numerical tests and solve the stochastic Darcy problem with deterministic loading term $f(x) = x$ in the one dimensional domain $D = [0, 1]$, both for a Gaussian and exponential covariance function. Homogeneous Dirichlet boundary conditions are imposed on $\Gamma_D = \{0, 1\}$.

8.1. Analysis of the Taylor truncation error. Let $Y(\omega, x)$ be a stationary centered Gaussian random field with squared exponential covariance function

$$\mathrm{Cov}_Y(x_1, x_2) = \sigma^2 \ \mathrm{e}^{-\frac{\|x_1-x_2\|^2}{L^2}}, \quad (x_1, x_2) \in D \times D$$

where $0 < \sigma < 1$ and $L = 0.2$ are the standard deviation and the correlation length of $Y(\omega, x)$, respectively. Let us take a uniform discretization of the spatial domain $D = [0, 1]$ in $N_h = 10000$ intervals ($h = 1/N_h$). As a first step, we perform the truncated KL-expansion of $Y(\omega, x)$ with a tolerance $\mathsf{tol}_{\mathsf{KL}} = 10^{-4}$, so that $N = 11$ random variables are considered and 99% of the variance of the field is captured. Using the algorithm described in Section 6.2 we then compute the 4-th order approximation $\mathbb{E}\left[T^4 u\right]$ of the expected value $\mathbb{E}[u]$. A small tolerance $\mathsf{tol} = 10^{-10}$ is imposed in all computations related to the TT-format, such that the loss of accuracy due to recompression is negligible. As reference solution we consider the mean of u computed via the collocation method on an isotropic total degree Smolyak sparse grid with 265 Gauss-Hermite collocation points (see e.g. [3]), on the same spatial discretization ($N_h = 10000$), see Figure 5.

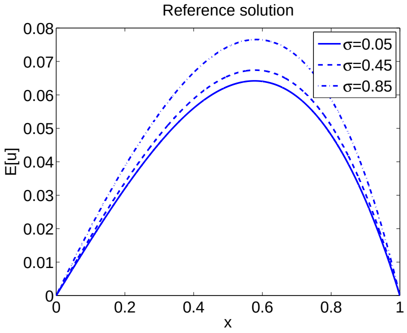

FIGURE 5. $\mathbb{E}[u]$ *computed via the collocation method, for different values of* σ.

Note that the error made by our method stems from different contributions: the truncation of the KL-expansion, approximations in the TT-format, the truncation of the Taylor series and the FE approximation. Here, we start from the same truncated KL-expansion both to compute the collocation solution and the TT solution; we use the same FE grid, and all computations in the TT-format are done with high precision ($\mathsf{tol} = 10^{-10}$). Hence, we observe only the error due to the truncation of the Taylor series.

In [5] we have shown that $\left\|\mathbb{E}[u] - \mathbb{E}\left[T^K u\right]\right\|_{L^2(D)} = O(\sigma^{K+1})$. Observe that, since $Y(\omega, x)$ is centered, all the odd-point correlations of Y vanish, so that

$$\left\|\mathbb{E}[u] - \mathbb{E}\left[T^K u\right]\right\|_{L^2(D)} = \begin{cases} O(\sigma^{K+2}) & \text{if } K \text{ is even,} \\ O(\sigma^{K+1}) & \text{if } K \text{ is odd.} \end{cases} \tag{8.1}$$

This theoretical bound is numerically confirmed in Figure 6 (left), where the computed error $\left\|\mathbb{E}[u] - \mathbb{E}\left[T^K u\right]\right\|_{L^2(D)}$ is plotted as a function of the standard deviation σ.

Figure 6 (right) shows the computed error $\left\|\mathbb{E}\left[u\right]-\mathbb{E}\left[T^K u\right]\right\|_{L^2(D)}$ as a function of K (at least up to $K=4$) for different values of σ. It turns out that for $\sigma<1$ it is always useful to take into account higher order corrections at least up to $K=4$. The maximum precision that the method can reach is 10^{-10}. We believe that this is due to the precision in the TT computations.

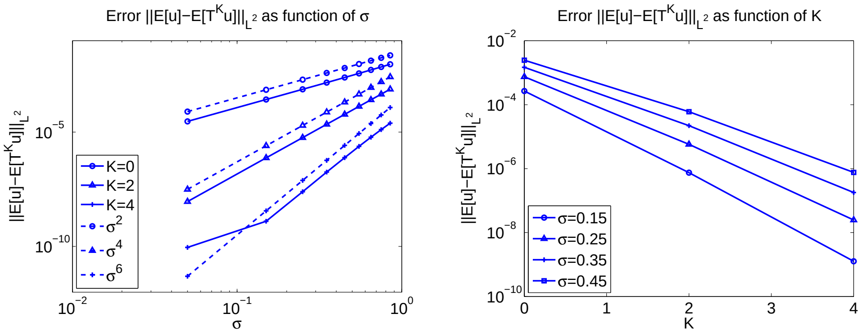

FIGURE 6. *Plot of the computed error* $\left\|\mathbb{E}\left[u\right]-\mathbb{E}\left[T^K u\right]\right\|_{L^2(D)}$ *as a function of* σ *(left) and* K *(right).*

Suppose now that $Y(\omega,x)$ is a conditioned field to N_{obs} available point-wise observations. In this case, the covariance function Cov_Y is non-stationary, but still Hölder continuous, so that the well-posedness of the stochastic Darcy problem still holds. Following the same steps as in Section 3, it is possible to derive the recursive first moment problem, which now involves the $(l+1)$-point correlations $\mathbb{E}\left[u^{k-l}\otimes(Y')^{\otimes l}\right]$, where $u^{k-l}:=D^{k-l}u(\mathbb{E}\left[Y\right])[Y']^{k-l}$ is the Gateaux derivative of u in $\mathbb{E}\left[Y\right]$ evaluated along the vector $\underbrace{(Y',\dots,Y')}_{k-l\text{ times}}$, and $Y'(\omega,x)$ is the centered Gaussian random field $Y'(\omega,x):=Y(\omega,x)-\mathbb{E}\left[Y\right](x)$.

Considering the case of $Y(\omega,x)$ conditioned to available observations is very relevant in applications. Indeed, suppose the domain D contains an heterogeneous porous medium. Although it is not possible to know its permeability everywhere, from the practical point of view it is possible to measure it in a certain number of fixed points. Hence, the natural model considered in the geophysical literature describes the permeability as a conditioned lognormal random field. See e.g. [27, 17, 18]. The more observations are available, the smaller the total variance of the field will be. This, actually, favors the use of perturbation methods.

As in the previous numerical test, let $N_h=10000$ be the number of subintervals of D, and $\mathsf{tol}=10^{-4}$ the tolerance imposed in the truncation of the KL-expansion. The N_{obs} observations available are evenly distributed in $D=[0,1]$. To capture the 99% of variability of the field, $N=9$ and $N=8$ random variables are needed in the cases $N_{obs}=3$ and $N_{obs}=5$ respectively. Note that, the higher the number of observations is, the smaller is the number of random variables needed to reach the same level of accuracy in the KL-expansion. The reference solution is computed via the collocation method (isotropic total degree Smolyak sparse grid with 181 and 145 Gauss-Hermite collocation points for $N_{obs}=3$ and $N_{obs}=5$, respectively), see Figure 7.

Figure 8 shows the behavior of the error $\left\|\mathbb{E}\left[u\right]-\mathbb{E}\left[T^K u\right]\right\|_{L^2(D)}$ as a function of σ, with $N_{obs}=3$ (left) and $N_{obs}=5$ (right). The same rate as for $N_{obs}=0$ (see (8.1)) is observed. In Figure 9 we plot the error as a function of K. The error is about

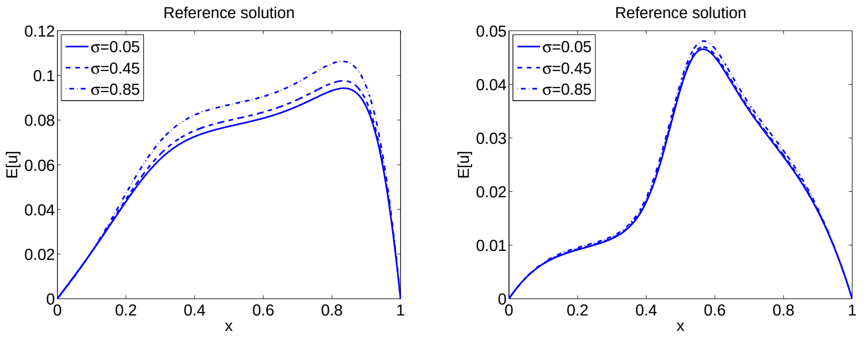

FIGURE 7. $\mathbb{E}[u]$ *computed via the collocation method, with* $N_{obs} = 3$ *(left) and* $N_{obs} = 5$ *(right).*

1 order of magnitude smaller for $N_{obs} = 3$ (compared to $N_{obs} = 0$) and 2 orders of magnitude smaller for $N_{obs} = 5$.

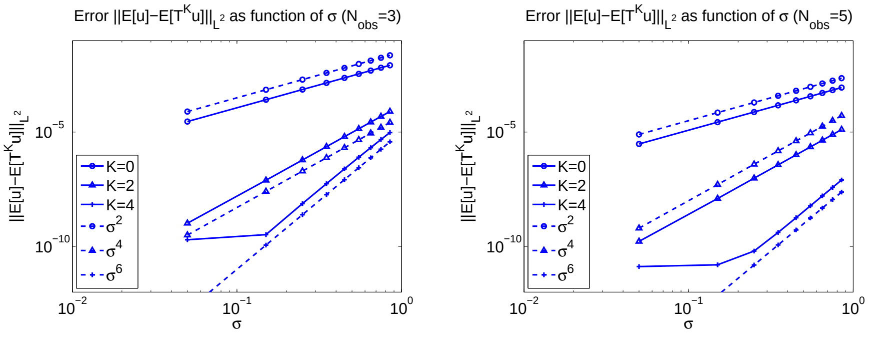

FIGURE 8. *Plot of the computed error* $\left\|\mathbb{E}[u] - \mathbb{E}\left[T^K u\right]\right\|_{L^2(D)}$ *as a function of* σ*, with* $N_{obs} = 3$ *(left) and* $N_{obs} = 5$ *(right).*

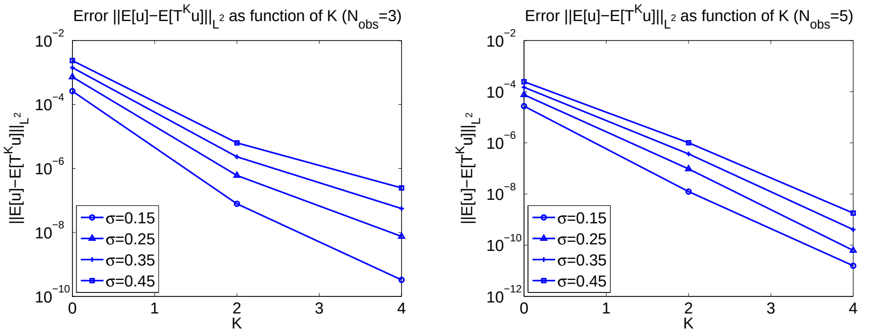

FIGURE 9. *Plot of the computed error* $\left\|\mathbb{E}[u] - \mathbb{E}\left[T^K u\right]\right\|_{L^2(D)}$ *as a function of* K*, with* $N_{obs} = 3$ *(left) and* $N_{obs} = 5$ *(right).*

8.2. Influence of the TT-format recompression. Let us consider a stationary Gaussian random field $Y(\omega, x)$ with squared exponential covariance function of correlation length $L = 0.2$. Let $N_h = 100$. Instead of truncating the KL-expansion

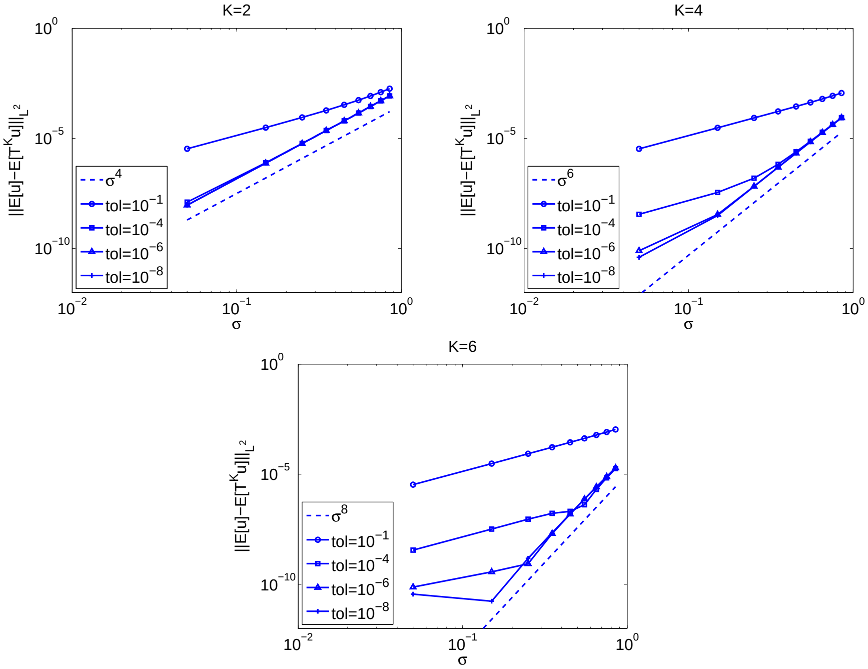

FIGURE 10. *Logarithmic plot of the computed error* $\left\|\mathbb{E}\left[u\right]-\mathbb{E}\left[T^{K}u\right]\right\|_{L^2(D)}$ *as a function of the standard deviation* σ*, for different tolerances.*

as done before, we consider the complete KL, that is we compute the expansion up to machine precision ($\mathsf{tol}_{\mathsf{KL}}=10^{-16}$). To reach this accuracy, $N=26$ random variables have to be considered. We run our implementation imposing different tolerances in the function **compute_moment_Y**. In this way, we can observe how the error $\left\|\mathbb{E}\left[u\right]-\mathbb{E}\left[T^{K}u\right]\right\|_{L^2(D)}$ depends on the approximation of $\mathbb{E}\left[Y^{\otimes K}\right]$ in the TT-format.

Figure 10 shows the computed error $\left\|\mathbb{E}\left[u\right]-\mathbb{E}\left[T^{K}u\right]\right\|_{L^2(D)}$ as a function of the standard deviation σ, for different tolerances, with $K=2,4,6$. A tolerance 10^{-1} is clearly too large; the predicted behavior (8.1) is not observed, even for $K=2$. In contrast, the tolerance 10^{-8} guarantees the predicted behavior for $K=2$, $K=4$ and $K=6$.

In Figure 11 we plot the error as a function of K, for different tolerances, with $\sigma=0.05$ (left), $\sigma=0.25$ (right) and $\sigma=0.85$ (down). The total error is the sum of two contributions: the truncation of the Taylor expansion and the tolerance used in the TT-format, which should ideally be balanced. In Figure 11 we see that, the smaller σ is, the smaller the tolerance in the TT-format computations has to be to equilibrate the truncation error.

Now, we attempt to investigate the complexity of our approach, i.e. how the total error depends on the computational cost of the algorithm. In particular, we numerically study the dependance of the error on the computational cost under the assumption that the computational cost of the recursive algorithm is dominated by the number M_1' of linear systems we have to solve in the recursion.

Figure 12 shows the logarithmic plot of the error $\left\|\mathbb{E}\left[u\right]-\mathbb{E}\left[T^{K}u\right]\right\|_{L^2(D)}$ as a function of the complexity M_1' for different tolerances in the function **compute_moment_Y**,

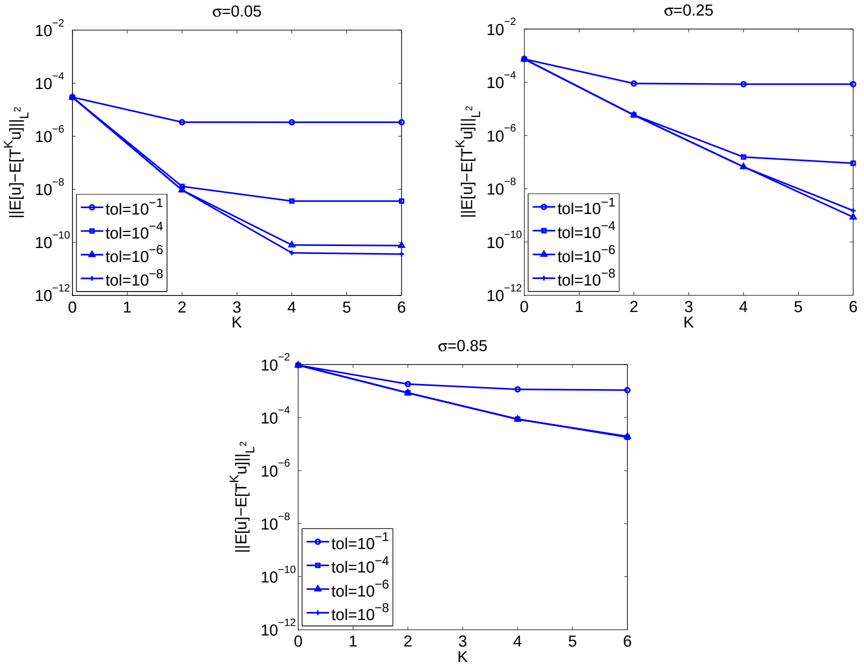

FIGURE 11. *Semilogarithmic plot of the computed error* $\left\| \mathbb{E}\left[u\right] - \mathbb{E}\left[T^K u\right] \right\|_{L^2(D)}$ *as a function of* K*, for different tolerances.*

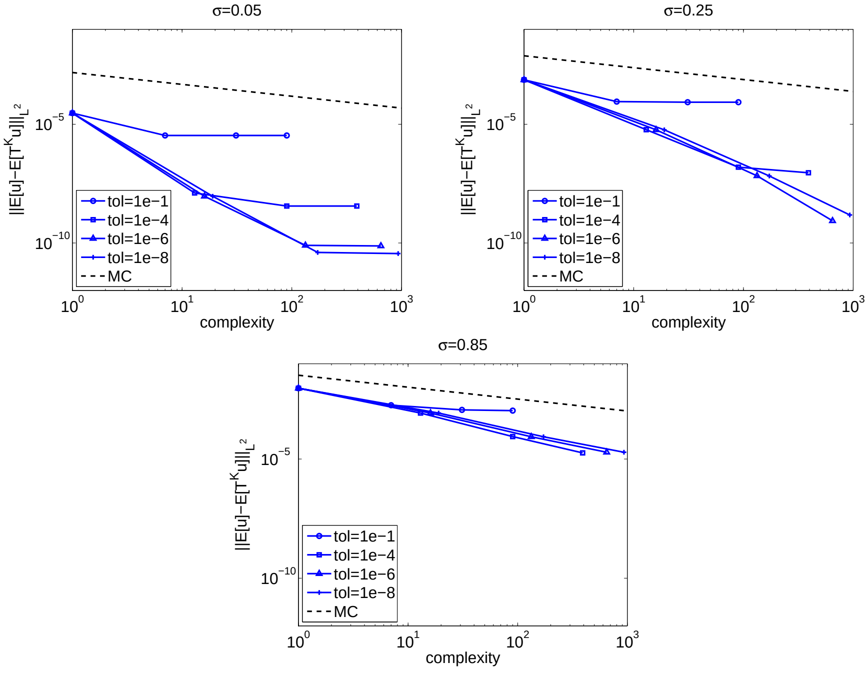

FIGURE 12. *Logarithmic plot of the computed error* $\left\| \mathbb{E}\left[u\right] - \mathbb{E}\left[T^K u\right] \right\|_{L^2(D)}$ *as a function of the computational cost (number of linear systems to solve) for different tolerances in the TT-format computations. The dashed black line gives an idea of the behavior a Monte Carlo estimator.*

with $\sigma = 0.05, 0.25, 0.85$. To give an idea of the behavior of the Monte Carlo estimate, we compare the computed error with the quantity $\dfrac{\sigma_{MC}}{\sqrt{M_1'}}$ (dashed black line), where $\sigma_{MC} = \left\| \left(\mathbb{E}\left[u - \mathbb{E}\left[u\right]\right]^2 \right)^{1/2} \right\|_{L^2(D)}$ has been estimated by the Monte Carlo method with 10000 samples. Note that, for small σ (e.g. $\sigma = 0.05$), the smaller the tolerance imposed is, the higher the accuracy reached. This is not the case if we let σ grow. Indeed, the error from the TT-format is not anymore the most influencing component of the error, which is dominated, instead, by the truncation of the Taylor series. For a fixed truncation level, there is therefore an *optimal choice* $\mathsf{tol}_{\mathsf{opt}}$ *of the tolerance*, depending both on σ and K. Figure 12 shows that, if the optimal tolerance is chosen, the performance of the moment equations is far superior to a standard Monte Carlo method. The question of how to determine a priori the optimal tolerance as a function of K and σ is still open and under investigation.

8.3. Exponential covariance function: a comparison with Monte Carlo.

In the previous numerical examples we have considered a very smooth random field with squared exponential covariance. We show in this section the performance of our algorithm for a non-smooth random field.

Let $Y(\omega, x)$ be a stationary centered Gaussian random field with exponential covariance function

$$\mathrm{Cov}_Y(x_1, x_2) = \sigma^2 \, \mathrm{e}^{-\frac{\|x_1 - x_2\|}{L}}, \quad (x_1, x_2) \in D \times D$$

with $0 < \sigma < 1$, $L = 0.2$ and $D = [0, 1]$. Let $N_h = 10000$. We compute the KL-expansion with tolerance $\mathsf{tol} = 10^{-4}$, so that $N = 6429$ random variables are considered and nearly the 100% of variance of the field is captured. Then, we compute the second order correction of $\mathbb{E}[u]$ with our proposed method. Since $N = 6429$, a collocation method becomes unfeasible. We compare, therefore, our method with a Monte Carlo simulation with $M = 10000$ samples. See Figure 13, where the second order correction is compared with the Monte Carlo method for $\sigma = 0.05$ (left) and $\sigma = 0.25$ (right). In plotting the Monte Carlo solution, we have also added error bars representing $\pm\sigma_{MC}$, where σ_{MC} is the estimated standard deviation of the Monte Carlo estimator. We observe that the TT-solution is always contained in the confidence interval of the Monte Carlo solution.

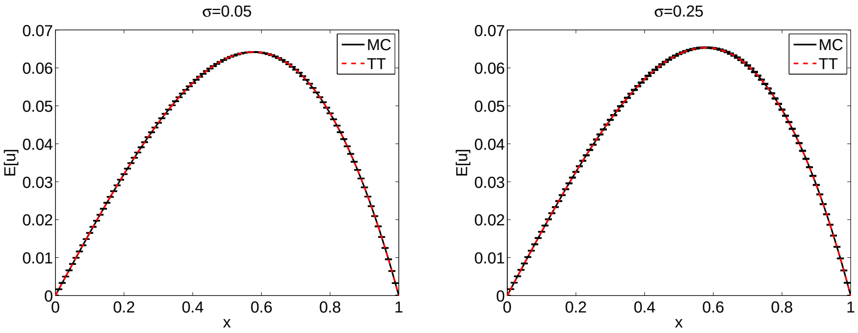

Figure 13. *Second order correction computed via our proposed method, and* $\mathbb{E}[u]$ *computed via the Monte Carlo method (*$M = 10000$ *samples) for* $\sigma = 0.05$ *(left) and* $\sigma = 0.25$ *(right). The TT solution compares well with the Monte Carlo solution.*

9. Conclusions. We have derived the full tensor product finite element formulation of the recursive problem solved by the $(l+1)$-point correlation functions $\mathbb{E}\left[Y^{\otimes l} \otimes u^{k-l}\right]$, for $k = 0, \ldots, K$ and $l = 0, \ldots, k$. Since the number of entries of a tensor is exponential in its order, we have introduced a low rank format (the TT-format) to store the tensors and make computations. We have developed an algorithm in TT-format that computes the K-th order approximation $\mathbb{E}\left[T^K u\right]$.

We have studied the storage requirements of our algorithm. The parameter we have taken into account is the TT-rank. We have performed some numerical tests to understand how the TT-rank of the input correlations $\mathcal{C}_k^{\mathrm{TT}}$ $(k = 0, \ldots, K)$ depends on the precision of the TT computations. Moreover, we have shown the evolution of the TT-rank along the recursion, i.e. along each diagonal of Table 1.

We have run our code both in the case of an unconditioned and conditioned Gaussian random field $Y(\omega, x)$. The more observations are available, the less the variability of the field is, so that the use of perturbation methods is favorable.

We have numerically studied how the error $\left\|\mathbb{E}\left[u\right] - \mathbb{E}\left[T^K u\right]\right\|_{L^2(D)}$ depends on both the truncation of the Taylor series, the tolerance imposed in the TT computations and the standard deviation σ of the Gaussian random field Y.

When a squared exponential covariance function is considered, we have compared our approach with the collocation method. On the other hand, when the collocation method in unfeasible (e.g. if an exponential covariance function is considered), our algorithm still provides a valid solution, which we have qualitatively compared with a standard Monte Carlo solution.

In this paper only one-dimensional computations have been performed. Nevertheless, we emphasize that spatial discretizations up to $N_h = 10000$ elements are considered. Hence, we believe that our algorithm can also easily deal with two-dimensional domains.

REFERENCES

[1] B. W. Bader, T. G. Kolda, et al. Matlab tensor toolbox version 2.5. Available online, January 2012.

[2] J. Bear and A. H.-D. Cheng. *Modeling Groundwater Flow and Contaminant Transport*, volume 23. Springer Netherlands, 2010.

[3] J. Beck, F. Nobile, L. Tamellini, and R. Tempone. A quasi-optimal sparse grids procedure for groundwater flows. In *Spectral and High Order Methods for Partial Differential Equations - ICOSAHOM 2012*, volume 95 of *Lecture Notes in Computational Science and Engineering*, pages 1–16. Springer International Publishing, 2014.

[4] F. Bonizzoni. *Analysis and approximation of moment equations for PDEs with stochastic data.* PhD thesis, Department of Mathematics, Politecnico di Milano, Italy, 2013.

[5] F. Bonizzoni and F. Nobile. Perturbation analysis for the Darcy problem with log-normal permeability. *SIAM/ASA Journal on Uncertainty Quantification*, 2(1):223–244, 2014.

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

[7] J. D. Carroll and J. J. Chang. Analysis of individual differences in multidimensional scaling via an n-way generalization of "eckart-youn" decomposition. *Psychometrika*, 35:283–319, 1970.

[8] J. Charrier and A. Debussche. Weak truncation error estimates for elliptic pdes with lognormal coefficients. *Stochastic Partial Differential Equations: Analysis and Computations*, 1(1):63–93, 2013.

[9] A. Chernov. Sparse polynomial approximation in positive order Sobolev spaces with bounded mixed derivatives and applications to elliptic problems with random loading. *Appl. Numer. Math.*, 62(4):360–377, 2012.

[10] A. Chernov and C. Schwab. First order k-th moment finite element analysis of nonlinear operator equations with stochastic data. *Math. Comp.*, 82:1859–1888, 2013.

[11] A. Chkifa, A. Chohen, and R. DeVore. Sparse adaptive Taylor approximation algorithms for

parametric and stochastic elliptic PDES. *ESAIM: Mathematical Modelling and Numerical Analysis*, 47:253–280, 2013.

[12] A. Cohen, R. Devore, and C. Schwab. Analytic regularity and polynomial approximation of parametric and stochastic elliptic PDE's. *Anal. Appl. (Singap.)*, 9(1):11–47, 2011.

[13] G. Dagan. *Flow and Transport in Porous Formations.* Springer-Verlag Heidelberg Berlin New York, 1989.

[14] J. Galvis and M. Sarkis. Approximating infinity-dimensional stochastic Darcy's equations without uniform ellipticity. *SIAM J. Numer. Anal.*, 47(5):3624–3651, 2009.

[15] C. J. Gittelson. Stochastic Galerkin discretization of the log-normal isotropic diffusion problem. *Math. Models Methods Appl. Sci.*, 20(2):237–263, 2010.

[16] L. Grasedyck. Hierarchical singular value decomposition of tensors. *SIAM J. Matrix Anal. Appl.*, 31(4):2029–2054, 2009/10.

[17] A. Guadagnini and S. P. Neuman. Nonlocal and localized analyses of conditional mean steady state flow in bounded, randomly nonuniform domains: 1. Theory and computational approach. *Water Resour. Res.*, 35(10):2999–3018, 1999.

[18] A. Guadagnini and S. P. Neuman. Nonlocal and localized analyses of conditional mean steady state flow in bounded, randomly nonuniform domains: 2. Computational examples. *Water Resour. Res.*, 35(10):3019–3039, 1999.

[19] W. Hackbusch. *Tensor spaces and numerical tensor calculus*, volume 42. Springer, 2012.

[20] W. Hackbusch and S. Kühn. A new scheme for the tensor representation. *Journal of Fourier Analysis and Applications*, 15:706–722, 2009.

[21] R. A. Harshman. Foundations of the parafac procedure: Models and conditions for an "explanatory" multimodal factor analysis. *UCLA Working Papers in Phonetics*, 16:1 – 84, 1970.

[22] T. Kolda and B. Bader. Tensor decompositions and applications. *SIAM Review*, 51(3):455–500, 2009.

[23] D. Kressner, R. Kumar, F. Nobile, and C. Tobler. Low-rank tensor approximation for high-order correlation functions of Gaussian random fields. Technical Report 24.2014, MATHICSE, École Polytechnique Fédérale de Lausanne, Lausanne, May 2014.

[24] D. Kressner and C. Tobler. Algorithm 941: `htucker`–a Matlab toolbox for tensors in hierarchical Tucker format. *ACM Trans. Math. Software*, 40(3):Art. 22, 22, 2014.

[25] I. V. Oseledets. MATLAB TT-Toolbox Version 2.2, May 2011. Available at `http://spring.inm.ras.ru/osel/?page_id=24`.

[26] I. V. Oseledets. Tensor-train decomposition. *SIAM J. Sci. Comput.*, 33(5):2295–2317, 2011.

[27] M. Riva, A. Guadagnini, and M. De Simoni. Assessment of uncertainty associated with the estimation of well catchments by moment equations. *Advances in Water Resources*, 29(5):676 – 691, 2006.

[28] R. Schneider and A. Uschmajew. Approximation rates for the hierarchical tensor format in periodic Sobolev spaces. *J. Complexity*, 30(2):56–71, 2014.

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

[30] D. M. Tartakovsky and S. P. Neuman. Transient flow in bounded randomly heterogeneous domains: 1. Exact conditional moment equations and recursive approximations. *Water Resour. Res.*, 34(1):1–12, 1998.

[31] C. Tobler. Matlab software package `constr_tt`, 2013. Available from `http://anchp.epfl.ch/software/misc.`

[32] R. A. Todor. *Sparse Perturbation Algorithms for Elliptic PDE's with Stochastic Data.* PhD thesis, ETH Zurich, Switzerland, 2005.

[33] L. R. Tucker. Some mathematical notes on three-mode factor analysis. *Psychometrika*, 31:279–311, 1966.

Recent publications:

MATHEMATICS INSTITUTE OF COMPUTATIONAL SCIENCE AND ENGINEERING
Section of Mathematics
Ecole Polytechnique Fédérale
CH-1015 Lausanne

26.2014 Abdul-Lateef Haji-Ali, Fabio Nobile, Raúl Tempone:
Multi index Monte Carlo: when sparsity meets sampling

27.2014 Assyr Abdulle, Orane Jecker:
An optimization-based multiscale coupling method

28.2014 Laura Iapichino, Alfio Quarteroni, Gianluigi Rozza, Stefan Volkwein:
Reduced basis method for the stokes equations in decomposable parametrized domains using greedy optimization

29.2014 Assyr Abdulle, Patrick Henning:
Localized orthogonal decomposition method for the wave equation with a continuum of scales

30.2014 Daniel Kressner, André Uschmajew:
On low-rank approximability of solutions to high-dimensional operator equations and eigenvalue problems

31.2014 Assyr Abdulle, Martin Huber:
Finite element heterogeneous multiscale method for nonlinear monotone parabolic homogenization problems

32.2014 Assyr Abdulle, Martin Huber, Gilles Vilmart:
Linearized numerical homogenization method for nonlinear monotone parabolic multiscale problems

33.2014 Marco Discacciati, Paola Gervasio, Alfio Quarteroni:
Interface control domain decomposition (ICDD) methods for heterogeneous problems

34.2014 Andre Uschmajew:
A new convergence proof for the high-order power method and generalizations

35.2014 Assyr Abdulle, Ondrej Budác:
A Petrov-Galerkin reduced basis approximation of the Stokes equation in parametrized geometries

36.2014 Assyr Abdulle, Martin E. Huber:
Error estimates for finite element approximations of nonlinear monotone elliptic problems with application to numerical homogenization

37.2014 Lars Karlsson, Daniel Kressner, André Uschmajew:
Parallel algorithms for tensor completion in the CP format

38.2014 Paolo Tricerri, Luca Dedè, Simone Deparis, Alfio Quarteroni, Anne M. Robertson, Adélia Sequeira:
Fluid-structure interaction siumlations of cerebral arteries modeled by isotropic and anistropic constitutive laws

39.2014 Francesca Bonizzoni, Fabio Nobile, Daniel Kressner:
Tensor train approximation of moment equations for the log-normal Darcy problem