A SPARSE FFT APPROACH FOR ODE WITH RANDOM COEFFICIENTS

MAXIMILIAN BOCHMANN*, LUTZ KÄMMERER†, AND DANIEL POTTS‡

Abstract. The paper presents a general strategy to solve ordinary differential equations (ODE), where some coefficient depend on the spatial variable and on additional random variables. The approach is based on the application of a recently developed dimension-incremental sparse fast Fourier transform. Since such algorithms require periodic signals, we discuss periodization strategies and associated necessary deperiodization modifications within the occuring solution steps.

The computed approximate solutions of the ODE depend on the spatial variable and on the random variables as well. Certainly, one of the crucial challenges of the high dimensional approximation process is to rate the influence of each variable on the solution as well as the determination of the relations and couplings within the set of variables. The suggested approach meets these challenges in a full automatic manner with reasonable computational costs, i.e., in contrast to already existing approaches, one does not need to seriously restrict the used set of ansatz functions in advance.

Keywords and phrases : ordinary differential equation with random coefficient, sparse fast Fourier transform, sparse FFT, lattice FFT, lattice rule, periodization, uncertainty quantification, approximation of moments, high dimensional approximation

2010 AMS Mathematics Subject Classification : 42A10, 60H10, 60H35, 65C20, 65N35, 65T40, 65T50

1. Introduction. During the last years, the concept of random variables has become a very popular tool to model uncertain properties mathematically. For instance, diffusion characteristics of inhomogeneous materials can be distinctly more accurately described by functions that additionally depend on random variables. One common area of application of these mathematical designs are diffusion coefficients in differential equations. Certainly, the additional random variables affect the solvability and – if exist – the solutions of the differential equations under consideration. Besides investigations on existence, uniqueness and regularity of solutions for specific mathematical problems that involve randomness, cf. e.g. [5, 10, 2, 1], numerical solution approaches need to be developed in order to compute approximations of the desired solutions. Accordingly, the established numerical solution approaches for differential equations without random coefficients need to be – at least – extended in order to meet the new challenges that are caused by the randomness of the diffusion coefficient. Commonly, discretizations of the domain of the stochastic variables lead to discretized solutions that are used to compute solutions in polynomial spaces or finite element representations, cf. e.g. [16, 8, 19, 9, 7]. One essential task in this approach is the choice of suitable polynomial spaces and corresponding basis polynomials, which can be extremely challenging for higher numbers of random variables and occuring dependencies within the random variables. Furthermore, preferable choices of the used basis functions can improve the efficiency of the arising computations.

In the recent literature, many solution approaches deal with differential equations with random parameters in their coefficients. Most commonly, one suggests to choose several fixed instances of the random variables and applies known solvers for the considered differential equations without random

*Chemnitz University of Technology, Faculty of Mathematics, 09107 Chemnitz, Germany, maximilian.bochmann@s2014.tu-chemnitz.de

†Chemnitz University of Technology, Faculty of Mathematics, 09107 Chemnitz, Germany, kaemmerer@mathematik.tu-chemnitz.de, Phone:+49-371-531-37728, Fax:+49-371-531-837728

‡Chemnitz University of Technology, Faculty of Mathematics, 09107 Chemnitz, Germany, potts@mathematik.tu-chemnitz.de, Phone:+49-371-531-32150, Fax:+49-371-531-832150

coefficients for each of those instances. One achieves a set of solutions and computes the quantities of interest, which may be the coefficients of a specific expansion of the full solution of the differential equation or simply the expectation function, from these solutions by applying stochastic estimators.

In this paper, we present a closed approach that deals with the spatial variable as well as the random variables simultaneously in order to solve an ordinary differential equation with a diffusion coefficient affected by randomness. In more detail, we consider the differential equation

$$(1.1) \qquad -\frac{\partial}{\partial \eta}\left(a(\eta,\boldsymbol{\xi})\frac{\partial}{\partial \eta}u(\eta,\boldsymbol{\xi})\right) = f(\eta), \qquad u(\alpha,\boldsymbol{\xi}) = u(\beta,\boldsymbol{\xi}) = 0$$

with homogeneous boundary conditions, where $a\colon D_a \to \mathbb{R}$, $D_a := \bigtimes_{j=1}^{1+d_{\boldsymbol{\xi}}}[\alpha_j,\beta_j] \subset \mathbb{R} \times \mathbb{R}^{d_{\boldsymbol{\xi}}}$, is the diffusion coefficient, that depends on the spatial variable η as well as the random variables that are the components of the vector $\boldsymbol{\xi} \in \mathbb{R}^{d_{\boldsymbol{\xi}}}$, and the right hand side $f\colon D_f \to \mathbb{R}$, $D_f := [\alpha_1,\beta_1] \subset \mathbb{R}$, is a function that depends only on the spatial variable η. Here we would like to point out, that the suggested solution approach is not restricted to the homogeneous boundary conditions or the independence on the random variables $\boldsymbol{\xi}$ of the right hand side f. Simple modifications of the presented approach leads to a solution strategy even for more general settings. However, the restrictions will simplify the notations and help to preserve clarity.

The essential restriction

$$(1.2) \qquad 0 < r \le a(\eta,\boldsymbol{\xi}) \le R < \infty$$

for all $(\eta,\boldsymbol{\xi}) \in D_a$ guarantees the existence of a unique solution $u(\circ,\boldsymbol{\xi})$ of (1.1) for each fixed $\boldsymbol{\xi}$. Hence, we suggest to approximately compute the unique solution u of (1.1) by means of a dimension-incremental sparse fast Fourier transform (FFT) approach, cf. [18, 15], and based on a direct reversion of the occurring derivatives.

On the one hand, the assumptions on the differential equation (1.1) do not guarantee for periodic signals that has to be treated. On the other hand, the sparse FFT approaches consider the input signals as periodic signals and is more successful – in the sense of approximation rates, number of needed samples to ensure a specific accuracy, etc. – when dealing with smooth periodic signals. Therefore, the periodization of the arising signals will be necessary in order to compute good approximate solutions of (1.1). At this point we would like to highlight that we approximately compute a complete solution of (1.1), which can be used to subsequently approximate several quantities of interest. The crucial advantage of this approach is that these complete approximate solutions reveals detailed characteristics of the random variables, i.e., the influence of each single variable on the solution as well as the interaction between different variables. We stress that the suggested strategy automatically detects these detailed characteristics with reasonable computational costs.

The paper is organized as follows: First we roughly outline the concept of the dimension-incremental sparse FFT approach and indicate the basic properties of suitable periodization mappings in Section 2. Section 3 presents the suggested strategy to treat the considered problem which leads to an approximation of the solution of the considered differential equation. As mentioned above, one may be interested in specific quantities of interest of this solution. Thus, we demonstrate how to compute nth moments of the computed solution in Section 4. Section 5 contains various numerical examples, shows the operability of the suggested approach, and discusses advantages and disadvantages of the applied sparse FFT approaches.

2. Prerequisites.

2.1. Sparse FFT. As mentioned above, we suggest to approximate the solution of an ordinary differential equation with random coefficients using the dimension-incremental FFT approach presented in [15]. In this section, we declare the necessary notation and indicate the basic idea of this algorithm.

The aim of the dimension-incremental approach is the reconstruction of the Fourier coefficients $\hat{p}_{\boldsymbol{k}}$, $\boldsymbol{k} \in I$, of an arbitrarily chosen trigonometric polynomial

$$(2.1) \qquad p(\boldsymbol{x}) := \sum_{\boldsymbol{k} \in I} \hat{p}_{\boldsymbol{k}} \, \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{k} \cdot \boldsymbol{x}},$$

where the frequencies $\boldsymbol{k}$ are supported on a frequency set $I \subset \mathbb{Z}^d$ of finite cardinality, i.e. $|I| < \infty$. In contrast to usual FFT algorithms, the challenge of sparse FFT algorithms is the efficient determination of the unknown frequency set I in addition to the Fourier coefficients $\hat{p}_{\boldsymbol{k}}$ using only sampling values of p.

Appropriate thresholding strategies within dimension-incremental sparse FFT algorithms allow for the treatment of general functions $f \in L_1(\mathbb{T}^d) \cap \mathcal{C}(\mathbb{T}^d)$, i.e., the sparse FFT determines an approximation of the frequency index set I as well as an approximation of the (roughly) largest Fourier coefficients of the function f. Accordingly, the algorithms can be used in order to compute approximations

$$(2.2) \qquad \tilde{S}_I[f](\boldsymbol{x}) := \sum_{\boldsymbol{k} \in I} \hat{f}_{\boldsymbol{k}} \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{k} \cdot \boldsymbol{x}}$$

of the Fourier partial sum

$$S_I[f](\boldsymbol{x}) := \sum_{\boldsymbol{k} \in I} c_{\boldsymbol{k}}(f) \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{k} \cdot \boldsymbol{x}}$$

for sufficiently smooth functions f. In this context, the Fourier partial sum $S_I[f]$ is the truncated Fourier series of f, which implies the formal definition of the Fourier coefficients

$$c_{\boldsymbol{k}}(f) := \int_{\mathbb{T}^d} f(\boldsymbol{x}) \mathrm{e}^{-2\pi \mathrm{i} \boldsymbol{k} \cdot \boldsymbol{x}} \mathrm{d}\boldsymbol{x}.$$

In general, the coefficients $\hat{f}_{\boldsymbol{k}}$ are just approximations of the Fourier coefficients $c_{\boldsymbol{k}}(f)$, since they are computed using only function evaluations of f and thus are disturbed at least by aliasing.

In order to compute both, the set I of the most significant frequencies $\boldsymbol{k}$ as well as approximations $\hat{f}_{\boldsymbol{k}}$ of the corresponding Fourier coefficients, a dimension-incremental approach was developed in [18, 15], where the fundamental concept arises from a dimension-incremental method for the reconstruction of anharmonic trigonometric polynomials based on Prony's method, cf. [17]. An outline of this concept can be found in [15, Sec. 2.2].

In this paper, we restrict the discussion to the in- and output of the algorithm, cf. Algorithm 2.1. We require a restricted search space $\Gamma \subset \mathbb{Z}^d$ in frequency domain, where the significant Fourier coefficients are assumed to be supported. For simplicity and without crucial influence on the runtime of the algorithm, we can choose a tensor product box of equal edge lengths, i.e., we fix $\Gamma = [-N, N]^d$ for a suitable edge length $2N + 1$, $N \in \mathbb{N}$. Since the used sampling nodes are chosen adaptively, we assume the function f being given as a black box. The parameter $\theta \in \mathbb{R}^+$ is a thresholding for the minimal absolute values that should be accepted as significant Fourier coefficient $\hat{f}_{\boldsymbol{k}}$ and its projections in lower dimensions. Additional sparsity parameters $s, s_{\mathrm{local}} \in \mathbb{N}$ restrict the algorithm to deal with at most s or $r \, s_{\mathrm{local}}$ frequencies in each dimension-incremental step. Here, the parameter r is the number of projections that are used in each dimension-incremental step. Multiple projections are necessary in

Algorithm 2.1 Reconstruction of a multivariate function f from sampling values along (multiple) rank-1 lattices (sFFT).

Input:	$\Gamma \subset \mathbb{Z}^d$	search space in frequency domain
	$f(\circ) \colon \mathbb{T}^d \to \mathbb{C}$	function f as black box (function handle)
	$\theta \in \mathbb{R}^+$	relative threshold
	$s, s_{\text{local}} \in \mathbb{N}$	sparsity parameters ($s_{\text{local}} := s$ by default)
	$r \in \mathbb{N}$	number of detection iterations
	$b \in \mathbb{N}$	maximal number of multiple rank-1 lattice searches per dimension-incremental step

BLACK BOX ALGORITHM Available at [21], details in [18, Alg. 1].

Output:	$I \subset \Gamma \subset \mathbb{Z}^d$	index set of detected frequencies, $|I| \leq \min\{s, |\Gamma|\}$
	$\hat{\boldsymbol{f}} \in \mathbb{C}^{|I|}$	corresponding Fourier coefficients of $\tilde{S}_I f$, cf. (2.2)

order to avoid detection failures caused by cancellations. Since we will use only function evaluations of the function f in order to compute an approximation, we have to apply suitable sampling strategies. For the case where we use multiple rank-1 lattices, the adaptive construction of the sampling set is affected by a certain small default probability. Therefore, it may happen that one has to start this construction of the sampling set more than once. The parameter $b \in \mathbb{N}$ can be used in order to restrict the number of restarts of the construction in each dimension-incremental step in order to guarantee the termination of the algorithm, cf. [15]. However, this parameter is not restrictive during the computation, since even the choice $b = 5$ is not reached in practice.

The output of Algorithm 2.1 is the frequency set I and the corresponding approximated Fourier coefficients $\hat{f}_{\boldsymbol{k}}$, $\boldsymbol{k} \in I$, where $\tilde{S}_I f$, cf. (2.2), is a good approximation of f when all significant frequencies are collected in I.

One crucial point of the dimension-incremental approach is the construction of spatial discretizations for trigonometric polynomials with frequencies in a certain, adaptively determined candidate set. Additional preferable properties of these spatial discretizations are

- fast discrete Fourier transform algorithms,
- fast construction methods for the spatial discretizations, and
- low oversampling factors, i.e., the ratio of the number of sampling values to the cardinality of the candidate set should be low.

For high dimensional sparse trigonometric polynomials the concept of multiple rank-1 lattices, cf. [14, 13] combines all these advantages, that are particularly beneficial to our targeted application.

For the sake of completeness, we give further details on the used sampling schemes. For a given generating vector $\boldsymbol{z} \in \mathbb{Z}^d$ and a lattice size $M \in \mathbb{N}$, we define the rank-1 lattice

$$\Lambda(\boldsymbol{z}, M) = \left\{ \frac{j}{M} \boldsymbol{z} \bmod \boldsymbol{1} \colon j = 0, \ldots, M-1 \right\},$$

where the modulo operation is applied componentwise. For a given frequency set $I \subset \mathbb{Z}^d$, $|I| < \infty$, the corresponding Fourier matrix is given by

$$\boldsymbol{A} = \boldsymbol{A}(\Lambda(\boldsymbol{z}, M), I) := \left(e^{2\pi \mathrm{i} \boldsymbol{k} \cdot \boldsymbol{z} \frac{j}{M}} \right)_{j=0,\ldots,M-1, \boldsymbol{k} \in I}.$$

The dimension-incremental sparse FFT deals with different candidate sets of frequencies and asks for spatial discretizations for trigonometric polynomials with frequencies supported on these frequency sets. Additional requirements on $\Lambda(\boldsymbol{z}, M)$ guarantees the spatial discretization property, i.e., the full column rank of the matrix $\boldsymbol{A}(\Lambda(\boldsymbol{z}, M), I)$. Due to the structure of $\Lambda(\boldsymbol{z}, M)$, the computations of the matrix vector products involving $\boldsymbol{A}$ and its pseudo inverse can be performed by fast Fourier transform algorithms, cf. [11]. These fast algorithms as well as the component–by–component construction algorithms for the used spatial discretizations, cf. [12], are the essential building blocks for the dimension-incremental sparse FFT based on single rank-1 lattices as spatial discretizations, which we denote by R1LsFFT, cf. [18] for details on that approach.

A very similar approach is considered in [15], where the authors replaced the used sampling schemes by multiple rank-1 lattices, i.e., the spatial discretizations are constructed by the union of more than one rank-1 lattice, which provides – at least with high probability – asymptotically lower oversampling factors as well as much faster construction approaches for spatial discretizations. Furthermore, fast Fourier transform algorithms for the evaluation and the reconstruction of trigonometric polynomials were developed, cf. [13]. The corresponding dimension-incremental sparse FFT that uses these algorithms, i.e., the FFT algorithms as well as the construction algorithms for the spatial discretizations, is denoted by MR1LsFFT in the following. Recently a dimension-incremental sparse FFT based on random sampling were suggested, cf. [4].

The aforementioned dimension-incremental sparse FFT algorithms can be applied to periodic functions. Higher order smoothness of the treated functions often leads to smaller and thus preferable frequency sets I. Hence, we consider reasonable approaches to (smoothly) periodize non-periodic functions.

2.2. Periodization. The goal of a periodization is the approximation of a non-periodic function $f \colon [\alpha, \beta] \to \mathbb{C}$ using trigonometric polynomials that are naturally periodic and corresponding fast Fourier transform algorithms. Accordingly, we transform f to a periodic function $\tilde{g}$ using a variable transform that has the following features

$$
\begin{aligned}
&\varphi \colon [0,1] \mapsto [\alpha, \beta], \\
&\varphi(0) = \alpha, \\
&\varphi(1/2) = \beta, \\
&\varphi(1/2 - x) = \varphi(1/2 + x) \quad \text{for } x \in [0, 1/2], \\
&\varphi \text{ is continuous in } [0,1/2] \text{ and strictly increasing in } (0,1/2),
\end{aligned}
$$

(2.3)

i.e., $f(\varphi(x)) = \tilde{g}(x)$. In more detail, we are interested in approximations of the antiderivative of the function f, which leads to

$$
(2.4) \qquad F(t) = \int_\alpha^t f(\tau) \mathrm{d}\tau = \int_{\varphi^{-1}(\alpha)}^{\varphi^{-1}(t)} \tilde{g}(x) \varphi'(x) \mathrm{d}x = \int_0^{\varphi^{-1}(t)} \tilde{g}(x) \varphi'(x) \mathrm{d}x, \qquad t \in [\alpha, \beta].
$$

In order to compute $F(t)$, $t \in [\alpha, \beta]$, we are interested in suitable approximations of $\tilde{g}(x) \varphi'(x)$ for $x \in [0, 1/2]$, which we want to realize using trigonometric polynomials. There are two different approaches to realize the computation of F. One point of view is to approximate $\tilde{g}(x)$ and assume that φ' is constant almost everywhere in $[0, 1/2]$, which leads to the well known tent transform approach [6, 20]. A more general approach will require additional assumptions on φ in order to obtain periodic smoothness of $\tilde{g}\varphi'$ that allow for suitable periodic approximations. For our purposes it is enough to deal with periodizations in one dimension. In higher dimensional settings, i.e., periodizations applied
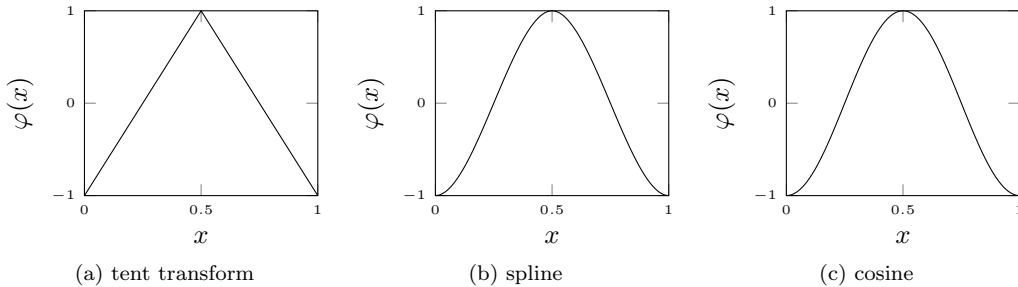
5

FIG. 2.1. *Different periodization mappings φ.*

to a vector of variables, we simply apply the one-dimensional periodizations to each component of the vector.

2.2.1. Tent transform. The so-called tent-transform [6, 20] is often used for periodization due to its simplicity. From a geometric point of view, the tent transform appends a mirror of the non-periodic function to the original function and dilates the resulting function such that its support is of length one. In addition, the new function is shifted such that its support is exactly $[0, 1]$. In formula, the mapping

$$\varphi \colon [0, 1] \to [\alpha, \beta], \qquad \varphi(x) = \beta - |2(\beta - \alpha)(1/2 - x)|$$

realizes this periodization of a function $f \colon [\alpha, \beta] \to \mathbb{C}$, cf. Figure 2.1a for a plot of φ, where $[\alpha, \beta] = [-1, 1]$. Certainly, this mapping φ is not continuously differentiable. Nevertheless, the constant first derivative φ' within $(0, 1/2)$ and $(1/2, 1)$ provides advantages within the integrals that we would like to deal with. For $t \in [\alpha, \beta]$ and $\varphi^{-1} \colon [\alpha, \beta] \to [0, 1/2]$, $\varphi^{-1}(t) = \frac{t - \alpha}{2(\beta - \alpha)}$, we obtain

$$\int_{\alpha}^{t} f(\tau)\mathrm{d}\tau = \int_{0}^{\varphi^{-1}(t)} f(\varphi(x))\varphi'(x)\mathrm{d}x = 2(\beta - \alpha) \int_{0}^{\frac{t - \alpha}{2(\beta - \alpha)}} f(\varphi(x))\mathrm{d}x.$$

Consequently, we only need to find an approximation of the antiderivative of the periodic function $f \circ \varphi$ in order to achieve an approximation of an antiderivative of the non-periodic function f.

2.2.2. More general periodizations. In addition to the basic assumptions on the peridization mapping, cf. (2.3), we may assume periodic differentiability in order to obtain smoother integrands in (2.4). Higher order smoothness of the periodization could have positive effects for the approximation of the integrand $\tilde{g}\varphi'$ using trigonometric polynomials. Roughly speaking, the smoother the function, the faster the decay of the Fourier coefficients, i.e, the smaller the cardinality of the frequency index set of suitable approximating trigonometric polynomials. In some cases it may be enough to construct periodizations of a specific fixed smoothness, since the function f does not allow for higher order smoothness of $\tilde{g}$. E.g., splines of higher order seems to be ideally suited in order to guarantee the desired properties, cf. Example 2.1. In cases of functions f of higher but unknown smoothness, infinitely differentiable mappings φ may be an option to ensure that the periodization does not cause lower order smoothness of the integrand $\tilde{g}\varphi'$. One suitable option for such a mapping is given in Example 2.2.

However, the usage of more complicated mappings may imply disadvantages in the computation of the inverse mapping of the periodization.

6

Example 2.1. A spline of order four can be used to construct a periodization that is two times continuously differentiable. The mapping φ, plotted in Figure 2.1b for $[\alpha, \beta] = [-1, 1]$, is given by

$$\varphi \colon [0, 1] \to [\alpha, \beta],$$

$$\varphi(x) = \begin{cases} -16(\beta - \alpha)x^3 + 12(\beta - \alpha)x^2 + \alpha & 0 \leq x \leq 1/2, \\ 16(\beta - \alpha)x^3 - 36(\beta - \alpha)x^2 + 24(\beta - \alpha)x + 5\alpha - 4\beta & 1/2 < x \leq 1. \end{cases}$$

Example 2.2. The cosine function can be used to construct a infinitely differentiable periodization mapping

$$\varphi \colon [0, 1] \to [\alpha, \beta], \qquad \varphi(x) = \frac{\alpha - \beta}{2} \cos(2\pi x) + \frac{\alpha + \beta}{2}.$$

A corresponding plot for $[\alpha, \beta] = [-1, 1]$ can be found in Figure 2.1c.

3. ODE solver. Since the differentiations within the ODE (1.1) acts on only one variable, we revert the differentiation by integration and thus obtain a formal solution

$$(3.1) \qquad u^*(t, \boldsymbol{\xi}) = \int_{\alpha_1}^t \frac{-\int_{\alpha_1}^\eta f(\tau)\mathrm{d}\tau + c_1(\boldsymbol{\xi})}{a(\eta, \boldsymbol{\xi})} \mathrm{d}\eta + c_2(\boldsymbol{\xi}).$$

However, for high-dimensional variables $\boldsymbol{\xi}$ the computation of such a solution is a particular challenge.

We denote by F the antiderivative of f and obtain $c_2(\boldsymbol{\xi}) = 0$ since $u^*(\alpha_1, \boldsymbol{\xi}) = 0$ for homogeneous boundary conditions. The solution $u^*(t, \boldsymbol{\xi})$ changes to

$$(3.2) \qquad u^*(t, \boldsymbol{\xi}) = \underbrace{\int_{\alpha_1}^t \frac{F(\alpha_1) - F(\eta)}{a(\eta, \boldsymbol{\xi})} \mathrm{d}\eta}_{:=u_1(t, \boldsymbol{\xi})} + c_1(\boldsymbol{\xi}) \underbrace{\int_{\alpha_1}^t \frac{1}{a(\eta, \boldsymbol{\xi})} \mathrm{d}\eta}_{:=u_2(t, \boldsymbol{\xi})},$$

where we will use the term $c_1(\boldsymbol{\xi})$ in order to satisfy the boundary condition $u^*(\beta_1, \boldsymbol{\xi}) = 0$. In particular, requirement (1.2) implies $u_2(\beta_1, \boldsymbol{\xi}) > 0$ and thus fixing

$$(3.3) \qquad c_1(\boldsymbol{\xi}) := -\frac{u_1(\beta_1, \boldsymbol{\xi})}{u_2(\beta_1, \boldsymbol{\xi})}$$

yields homogeneous boundary conditions for u^*. Accordingly, for given f and a we need to compute suitable approximations of u_1 and u_2. To this end, we will apply a dimension-incremental sparse FFT approach as described in Section 2.1. Since these FFT algorithms handles periodic signals, we need to periodize the upcoming functions.

3.1. Integration of the right hand side f. First, we determine the term $F(\alpha_1) - F(t)$ in (3.2) from above by approximating and integrating f. To this end, we periodize f using a suitable periodization φ, cf. Section 2.2,

$$\tilde{f}(x) = f(\varphi(x)).$$

Accordingly, we obtain

$$\int f(\tau)\mathrm{d}\tau = \int f(\varphi(x))\varphi'(x)\mathrm{d}x = \int \tilde{f}(x)\varphi'(x)\mathrm{d}x$$

7

and approximate the integrand on the right hand side by a trigonometric polynomial

$$(3.4) \qquad \tilde{S}_N\left[\tilde{f}\varphi'\right](x) = \sum_{k=-N}^{N} \hat{a}_k \mathrm{e}^{2\pi \mathrm{i}kx}.$$

An antiderivative of $\tilde{S}_N\left[\tilde{f}\varphi'\right]$ is given by

$$\int \tilde{S}_N\left[\tilde{f}\varphi'\right](x)\mathrm{d}x = \hat{a}_0 x + \sum_{1 \le |k| \le N} \frac{\hat{a}_k}{2k\pi\mathrm{i}} \mathrm{e}^{2\pi\mathrm{i}kx},$$

which yields

$$F(\tau) + c = \int f(\tau)\mathrm{d}\tau \approx \hat{a}_0 \varphi^{-1}(\tau) + \sum_{1 \le |k| \le N} \frac{\hat{a}_k}{2k\pi\mathrm{i}} \mathrm{e}^{2\pi\mathrm{i}k\varphi^{-1}(\tau)}.$$

Consequently, we denote the approximation of the term $F(\alpha_1) - F(t)$ by $\breve{F}(t)$ and obtain

$$F(\alpha_1) - F(\eta) \approx \breve{F}(\eta) := \hat{a}_0(\varphi^{-1}(\alpha_1) - \varphi^{-1}(\eta)) + \sum_{1 \le |k| \le N} \frac{\hat{a}_k}{2k\pi\mathrm{i}} \mathrm{e}^{2\pi\mathrm{i}k\varphi^{-1}(\alpha_1)} - \sum_{1 \le |k| \le N} \frac{\hat{a}_k}{2k\pi\mathrm{i}} \mathrm{e}^{2\pi\mathrm{i}k\varphi^{-1}(\eta)}$$

$$(3.5) \qquad = -\hat{a}_0\varphi^{-1}(\eta) - \sum_{1 \le |k| \le N} \frac{\hat{a}_k}{2k\pi\mathrm{i}}(\mathrm{e}^{2\pi\mathrm{i}k\varphi^{-1}(\eta)} - 1).$$

3.2. Approximating u_1, u_2, and c_1. We denote the integrands in (3.2) that determine u_1 and u_2 by v_1 and v_2, respectively, i.e. we have

$$v_1(\eta, \boldsymbol{\xi}) := \frac{F(\alpha_1) - F(\eta)}{a(\eta, \boldsymbol{\xi})} \qquad \text{and} \qquad v_2(\eta, \boldsymbol{\xi}) := \frac{1}{a(\eta, \boldsymbol{\xi})}.$$

First we consider the function v_1 and plug in the approximation $\breve{F}(\eta)$ of $F(\alpha_1) - F(\eta)$ from (3.5). This yields an approximation of v_1

$$\breve{v}_1(\eta, \boldsymbol{\xi}) := \frac{\breve{F}(\eta)}{a(\eta, \boldsymbol{\xi})}.$$

Now, our goal is to construct an antiderivative of $\breve{v}_1$ with respect to η, which is an approximation of the antiderivative of v_1. To this end, we construct a periodization of $\breve{v}_1$ using mappings φ_η and $\varphi_{\boldsymbol{\xi}}$. We take into account the influence of the periodization during integration with respect to the first variable, which leads to the periodic integrand $\tilde{\breve{v}}_1(x, \boldsymbol{y}) = \breve{v}_1(\varphi_\eta(x), \varphi_{\boldsymbol{\xi}}(\boldsymbol{y})) \varphi'_\eta(x)$. We compute a corresponding approximation

$$(3.6) \qquad \tilde{S}_{I_1}[\tilde{\breve{v}}_1](x, \boldsymbol{y}) := \sum_{(k, \boldsymbol{l}) \in I_1 \subset \mathbb{Z}^{1+d_{\boldsymbol{y}}}} \hat{b}_{(k, \boldsymbol{l})} \mathrm{e}^{2\pi\mathrm{i}(kx + \boldsymbol{l} \cdot \boldsymbol{y})},$$

which can be done by sparse FFT approaches, as described in Section 2.1, similar to those described in [18, 15]. The antiderivative of $\tilde{S}_{I_1}[\tilde{\breve{v}}_1]$ with respect to x is given by

$$(3.7) \qquad \int \tilde{S}_{I_1}[\tilde{\breve{v}}_1](x, \boldsymbol{y})\mathrm{d}x = \sum_{\substack{(k, \boldsymbol{l}) \in I_1 \\ k \ne 0}} \frac{\hat{b}_{(k, \boldsymbol{l})}}{2k\pi\mathrm{i}} \mathrm{e}^{2\pi\mathrm{i}(kx + \boldsymbol{l} \cdot \boldsymbol{y})} + x \sum_{(0, \boldsymbol{l}) \in I_1} \hat{b}_{(0, \boldsymbol{l})} \mathrm{e}^{2\pi\mathrm{i}\boldsymbol{l} \cdot \boldsymbol{y}} + C(\boldsymbol{y})$$

8

We choose $C(\boldsymbol{y}) := -\sum_{\substack{(k,\boldsymbol{l}) \in I_1 \\ k \neq 0}} \frac{\hat{b}_{(k,\boldsymbol{l})}}{2k\pi\mathrm{i}} \mathrm{e}^{2\pi\mathrm{i}\boldsymbol{l}\cdot\boldsymbol{y}}$ in order to guarantee $\breve{u}_1(\alpha_1, \boldsymbol{\xi}) = 0$ for all $\boldsymbol{\xi} \in D'_a :=$ $\times_{j=2}^{d_{\boldsymbol{\xi}}+1}[\alpha_j, \beta_j]$, and we roll the periodization back, which leads to the approximation

$$(3.8) \qquad \breve{u}_1(t, \boldsymbol{\xi}) := \sum_{\substack{(k,\boldsymbol{l}) \in I_1 \\ k \neq 0}} \underbrace{\frac{\hat{b}_{(k,\boldsymbol{l})}}{2k\pi\mathrm{i}}}_{=:\mathfrak{b}_{(k,\boldsymbol{l})}} \left(\mathrm{e}^{2\pi\mathrm{i}k\varphi_\eta^{-1}(t)} - 1\right) \mathrm{e}^{2\pi\mathrm{i}\boldsymbol{l}\cdot\varphi_{\boldsymbol{\xi}}^{-1}(\boldsymbol{\xi})} + \varphi_\eta^{-1}(t) \sum_{(0,\boldsymbol{l}) \in I_1} \underbrace{\hat{b}_{(0,\boldsymbol{l})}}_{=:\mathfrak{b}_{(0,\boldsymbol{l})}} \mathrm{e}^{2\pi\mathrm{i}\boldsymbol{l}\cdot\varphi_{\boldsymbol{\xi}}^{-1}(\boldsymbol{\xi})}$$

of u_1 given in (3.2).

The analogous approach, but without approximating v_2, leads to an approximation of $u_2(t, \boldsymbol{\xi})$

$$(3.9) \quad \breve{u}_2(t, \boldsymbol{\xi}) := \sum_{\substack{(k,\boldsymbol{l}) \in I_2 \subset \mathbb{Z}^{1+d_{\boldsymbol{y}}} \\ k \neq 0}} \underbrace{\frac{\hat{c}_{(k,\boldsymbol{l})}}{2k\pi\mathrm{i}}}_{=:\mathfrak{c}_{(k,\boldsymbol{l})}} \left(\mathrm{e}^{2\pi\mathrm{i}k\varphi_\eta^{-1}(t)} - 1\right) \mathrm{e}^{2\pi\mathrm{i}\boldsymbol{l}\cdot\varphi_{\boldsymbol{\xi}}^{-1}(\boldsymbol{\xi})} + \varphi_\eta^{-1}(t) \sum_{(0,\boldsymbol{l}) \in I_2} \underbrace{\hat{c}_{(0,\boldsymbol{l})}}_{=:\mathfrak{c}_{(0,\boldsymbol{l})}} \mathrm{e}^{2\pi\mathrm{i}\boldsymbol{l}\cdot\varphi_{\boldsymbol{\xi}}^{-1}(\boldsymbol{\xi})}.$$

The construction of $\breve{u}_j(t, \boldsymbol{\xi})$ yields $\breve{u}_j(\alpha_1, \boldsymbol{\xi}) = 0$, $j = 1, 2$. Consequently, each linear combination of $\breve{u}_1$ and $\breve{u}_2$ satisfies the homogeneous boundary condition in $t = \alpha_1$. A suitable approximation of $c_1(\boldsymbol{\xi})$, cf. (3.3), will lead to a linear combination of $\breve{u}_1$ and $\breve{u}_2$ that also satisfies the homogeneous boundary condition in $t = \beta_1$. To this end, we periodize $\breve{u}_1$ as well as $\breve{u}_2$ and construct the approximation

$$\check{c}_1(\boldsymbol{\xi}) := -\frac{\breve{u}_1(\beta_1, \boldsymbol{\xi})}{\breve{u}_2(\beta_1, \boldsymbol{\xi})} \qquad \text{and its periodization} \qquad \tilde{\check{c}}_1(\boldsymbol{y}) = -\frac{\breve{u}_1(\varphi_\eta(1/2), \varphi_{\boldsymbol{\xi}}(\boldsymbol{y}))}{\breve{u}_2(\varphi_\eta(1/2), \varphi_{\boldsymbol{\xi}}(\boldsymbol{y}))},$$

which are well defined due to the requirements on the diffusion coefficient a. We stress on the fact that the periodizations of $\breve{u}_j$ do not coincide to the terms in (3.7), since these are non-periodic in general due to the terms that are linear in x.

We approximate $\tilde{\check{c}}_1$ using sparse FFT approaches by

$$(3.10) \qquad\qquad \tilde{S}_{I_3}[\tilde{\check{c}}_1](\boldsymbol{y}) := \sum_{\boldsymbol{l} \in I_3 \subset \mathbb{Z}^{d_{\boldsymbol{y}}}} \mathfrak{d}_{\boldsymbol{l}} \mathrm{e}^{2\pi\mathrm{i}\boldsymbol{l}\cdot\boldsymbol{y}}.$$

and achieve an approximation of the non-periodic function c_1 by

$$(3.11) \qquad\qquad \check{c}_1(\boldsymbol{\xi}) = \sum_{\boldsymbol{l} \in I_3} \mathfrak{d}_{\boldsymbol{l}} \mathrm{e}^{2\pi\mathrm{i}\boldsymbol{l}\cdot\varphi_{\boldsymbol{\xi}}^{-1}(\boldsymbol{\xi})}.$$

Altogether, an approximation of $u^*(t, \boldsymbol{\xi})$, cf. (3.2), is then given by

$$(3.12) \qquad\qquad \breve{u}(t, \boldsymbol{\xi}) := \breve{u}_1(t, \boldsymbol{\xi}) + \check{c}_1(\boldsymbol{\xi})\,\breve{u}_2(t, \boldsymbol{\xi}),$$

which actually is built of three Fourier series combined with inverse mappings of the periodizations φ_η and $\varphi_{\boldsymbol{\xi}}$. Algorithm 3.1 summarizes the approach stated above.

3.3. Tent transform in spatial domain. For the specific choice of the tent transform, cf. Section 2.2.1, for the periodization φ_η and componentwise for $\varphi_{\boldsymbol{\xi}}$, we obtain some simplifications in the calculations above. Moreover, uniformly distributed random variables ξ_j lead to additional simplifications due to the constant probability density. We observe

$$\int_{\alpha_1}^\eta f(\tau)\mathrm{d}\tau = 2\int_0^{\frac{\eta-\alpha_1}{2(\beta_1-\alpha_1)}} f\big(\beta_1 - |2(\beta_1 - \alpha_1)(1/2 - x)|\big)\mathrm{d}x$$

9

Algorithm 3.1 Basic procedure for computing an approximation of the solution of (1.1) using a dimension-incremental sparse FFT approach

Input:	$f \colon \mathbb{T} \to \mathbb{C}$	function handle of right hand side f
	$a \colon \mathbb{T}^{1+d_{\boldsymbol{\xi}}} \to \mathbb{C}$	function handle of random coefficient a
	$\varphi_\eta \colon [0,1] \to [\alpha_1, \beta_1]$	periodization mapping of spatial variable η
	$\varphi_\eta^{-1} \colon [\alpha_1, \beta_1] \to [0, 1/2]$	inverse of the periodization mapping φ_η
	$\varphi_\eta' \colon [0,1] \to \mathbb{R}$	first derivative of φ_η
	$\varphi_{\boldsymbol{\xi}} \colon [0,1]^{d_{\boldsymbol{\xi}}} \to \times_{j=2}^{d_{\boldsymbol{\xi}}+1} [\alpha_j, \beta_j]$	periodization mapping of random variables $\boldsymbol{\xi}$
	$N \in \mathbb{N}, \theta \in \mathbb{R}, s \in \mathbb{N}$	sFFT parameters

1: Compute the Fourier coefficients $\{\hat{a}_k\}_{k=-N}^{N}$ of $\tilde{S}_N\left[\tilde{f}\varphi'\right]$
   by means of a 1d FFT using function values of $(f \circ \varphi_\eta)\varphi'_\eta$, cf. (3.4)
2: Compute the coefficients of the finite sum representation of non-periodic $\breve{F}$
   by modifying the coefficients $\{\hat{a}_k\}_{k=-N}^{N}$, cf. (3.5)
3: Compute the Fourier coefficients $\{\hat{b}_{(k,\boldsymbol{l})}\}_{(k,\boldsymbol{l}) \in I_1}$ of $\tilde{S}_{I_1}[\breve{\tilde{v}}_1]$ by means of an sFFT algorithm
   using sampling values of $\breve{F}(\varphi_\eta(x))\varphi'_\eta(x)/a(\varphi_\eta(x), \varphi_{\boldsymbol{\xi}}(\boldsymbol{y}))$, cf. (3.6)
4: Compute the Fourier coefficients $\{\hat{c}_{(k,\boldsymbol{l})}\}_{(k,\boldsymbol{l}) \in I_2}$ of $\tilde{S}_{I_2}[\breve{\tilde{v}}_2]$ by means of an sFFT algorithm
   using sampling values of $\varphi'_\eta(x)/a(\varphi_\eta(x), \varphi_{\boldsymbol{\xi}}(\boldsymbol{y}))$, similar to (3.6)
5: Compute the coefficients $\{\mathfrak{b}_{(k,\boldsymbol{l})}\}_{(k,\boldsymbol{l}) \in I_1}$ and $\{\mathfrak{c}_{(k,\boldsymbol{l})}\}_{(k,\boldsymbol{l}) \in I_2}$
   of the finite sum representation of non-periodic $\breve{u}_1$ and $\breve{u}_2$
   by modifying the coefficients $\{\hat{b}_{(k,\boldsymbol{l})}\}_{(k,\boldsymbol{l}) \in I_1}$ and $\{\hat{c}_{(k,\boldsymbol{l})}\}_{(k,\boldsymbol{l}) \in I_2}$, cf. (3.8) and (3.9)
6: Compute the Fourier coefficients $\{\mathfrak{d}_{\boldsymbol{l}}\}_{\boldsymbol{l} \in I_3}$ of $\tilde{S}_{I_3}[\breve{\tilde{c}}_1]$ by means of an sFFT algorithm
   using sampling values of $\breve{u}_1(\varphi_\eta(1/2), \varphi_{\boldsymbol{\xi}}(\boldsymbol{y}))/\breve{u}_2(\varphi_\eta(1/2), \varphi_{\boldsymbol{\xi}}(\boldsymbol{y}))$, cf. (3.10)

Output:	$\{\mathfrak{b}_{(k,\boldsymbol{l})}\}_{(k,\boldsymbol{l}) \in I_1}$	coefficients of $\breve{u}_1$, cf. (3.8)
	$\{\mathfrak{c}_{(k,\boldsymbol{l})}\}_{(k,\boldsymbol{l}) \in I_2}$	coefficients of $\breve{u}_2$, cf. (3.9)
	$\{\mathfrak{d}_{\boldsymbol{l}}\}_{\boldsymbol{l} \in I_3}$	coefficients of $\breve{\tilde{c}}_1$, cf. (3.11)

and, hence, it is enough to compute an approximation of $\tilde{f} = f \circ \varphi_\eta$, due to the equality

$$\tilde{S}_N[\tilde{f}\varphi'_\eta]\big|_{[0,1/2]} = 2\tilde{S}_N[\tilde{f}]\big|_{[0,1/2]}.$$

The approximation of $\tilde{S}_N[\tilde{f}]$ is preferable, since $\tilde{f}\varphi'_\eta$ is not continuous in the case where $f(\beta_1) \neq 0$ and, thus, problematic to approximate using trigonometric polynomials. Subsequent to the computation of $\tilde{S}_N[\tilde{f}] := \sum_{k=-N}^{N} \hat{a}_k \mathrm{e}^{2\pi \mathrm{i}kx}$, the calculations of the antiderivative and deperiodization leads to

$$\breve{F}(\eta) := 2 \left(-\hat{a}_0 \varphi^{-1}(\eta) + \sum_{1 \le |k| \le N} \frac{\hat{a}_k}{2k\pi\mathrm{i}} - \sum_{1 \le |k| \le N} \frac{\hat{a}_k}{2k\pi\mathrm{i}} \mathrm{e}^{2\pi \mathrm{i}k\varphi^{-1}(\eta)} \right)$$

$$= -\hat{a}_0 \frac{\eta - \alpha_1}{\beta_1 - \alpha_1} - \sum_{1 \le |k| \le N} \frac{\hat{a}_k}{k\pi\mathrm{i}}(\mathrm{e}^{\pi \mathrm{i}k \frac{\eta - \alpha_1}{\beta_1 - \alpha_1}} - 1).$$

The approximations $\breve{u}_1$ and $\breve{u}_2$, cf. (3.8) and (3.9), can be computed in the exact same manner. Altogether, using the tent transform in spatial domain requires slight modifications in Algorithm 3.1 in lines 1, 3, and 4. The used sampling values must be computed with a factor 2 instead of φ'_η.

4. Computing moments of the solution. In Section 3, we discussed a strategy for computing an approximate solution $\breve{u}$, cf. (3.12), of the ODE in (1.1). The computation of quantity of interests needs some further investigations. For simplicity, we demonstrate one approach to compute approximations of the nth moments of the solution u^* of the ODE in (1.1) based on the approximation $\breve{u}$. To this end, we denote the domain of the random variables by $D'_a := \times_{j=2}^{d_\xi+1}[\alpha_j, \beta_j]$.

The nth moment of the solution u^* of (1.1) is given by

$$u^*_{\mathrm{E}^n}(t) := \mathrm{E}((u^*(t, \circ))^n) = \int_{D'_a} (u^*(t, \boldsymbol{\xi}))^n \, \mathrm{d}\mu(\boldsymbol{\xi}) = \int_{D'_a} (u^*(t, \boldsymbol{\xi}))^n \, p(\boldsymbol{\xi}) \mathrm{d}\boldsymbol{\xi},$$

where p is the probability density function of the random variable vector $\boldsymbol{\xi}$. Periodization yields

$$\int_{D'_a} (u^*(t, \boldsymbol{\xi}))^n \, p(\boldsymbol{\xi}) \mathrm{d}\boldsymbol{\xi} = \int_{[0,1/2]^{d_\xi}} (u^*(\varphi_t(x), \varphi_{\boldsymbol{\xi}}(\boldsymbol{y})))^n \, p(\varphi_{\boldsymbol{\xi}}(\boldsymbol{y})) \, |\det(\boldsymbol{J})| \, \mathrm{d}\boldsymbol{y},$$

where $\boldsymbol{J}$ is the involved Jacobian matrix. Assuming $\varphi_{\boldsymbol{\xi}}$ is a periodization that acts on each component of $\boldsymbol{\xi}$ separately, cf. Section 2.2, the determinant of the Jacobian matrix is a tensor product function and we continue

$$= 2^{-d_{\boldsymbol{\xi}}} \int_{\mathbb{T}^{d_{\boldsymbol{\xi}}}} (u^*(\varphi_t(x), \varphi_{\boldsymbol{\xi}}(\boldsymbol{y})))^n \, p(\varphi_{\boldsymbol{\xi}}(\boldsymbol{y})) \prod_{j=1}^{d_{\boldsymbol{\xi}}} \left|\varphi'_{\xi_j}(y_j)\right| \mathrm{d}\boldsymbol{y}$$

(4.1)
$$\approx 2^{-d_{\boldsymbol{\xi}}} \int_{\mathbb{T}^{d_{\boldsymbol{\xi}}}} \underbrace{(\breve{u}(\varphi_t(x), \varphi_{\boldsymbol{\xi}}(\boldsymbol{y})))^n \, p(\varphi_{\boldsymbol{\xi}}(\boldsymbol{y})) \prod_{j=1}^{d_{\boldsymbol{\xi}}} \left|\varphi'_{\xi_j}(y_j)\right|}_{=:w_n(x,\boldsymbol{y})} \mathrm{d}\boldsymbol{y}.$$

We approximate the integrand w_n using a sparse FFT approach and achieve a Fourier partial sum

$$S_{I_4}[w_n](x, \boldsymbol{y}) = \sum_{(k,\boldsymbol{l}) \in I_4} \hat{a}_{(k,\boldsymbol{l})} \mathrm{e}^{2\pi\mathrm{i}(kx + \boldsymbol{l} \cdot \boldsymbol{y})}.$$

Integrating $S_{I_4}[w_n]$ instead of w_n in (4.1) leads to the approximation

$$\breve{u}^*_{\mathrm{E}^n}(t) := 2^{-d_{\boldsymbol{\xi}}} \int_{\mathbb{T}^{d_{\boldsymbol{\xi}}}} \sum_{(k,\boldsymbol{l}) \in I_w} \hat{a}_{(k,\boldsymbol{l})} \mathrm{e}^{2\pi\mathrm{i}(kx + \boldsymbol{l} \cdot \boldsymbol{y})} \mathrm{d}\boldsymbol{y} = 2^{-d_{\boldsymbol{\xi}}} \sum_{(k,\boldsymbol{0}) \in I_4} \hat{a}_{(k,\boldsymbol{0})} \mathrm{e}^{2\pi\mathrm{i}kx}$$

$$= 2^{-d_{\boldsymbol{\xi}}} \sum_{(k,\boldsymbol{0}) \in I_4} \hat{a}_{(k,\boldsymbol{0})} \mathrm{e}^{2\pi\mathrm{i}k\varphi_t^{-1}(t)}$$

of $u^*_{\mathrm{E}^n}$ since each monomial that depends on $\boldsymbol{y}$ integrates to zero.

5. Numerical results. For our numerical tests, we use an example from [3]. The goal is to numerically solve the boundary-value problem

(5.1)
$$-\frac{\partial}{\partial \eta}\left(a(\eta, \boldsymbol{\xi})\frac{\partial}{\partial \eta}u(\eta, \boldsymbol{\xi})\right) = 10, \text{ with } u \equiv 0 \text{ at } \partial(0,1),$$

where the random coefficient $a : [0,1] \times [-1,1]^{d_{\boldsymbol{\xi}}} \to \mathbb{R}$ is given by
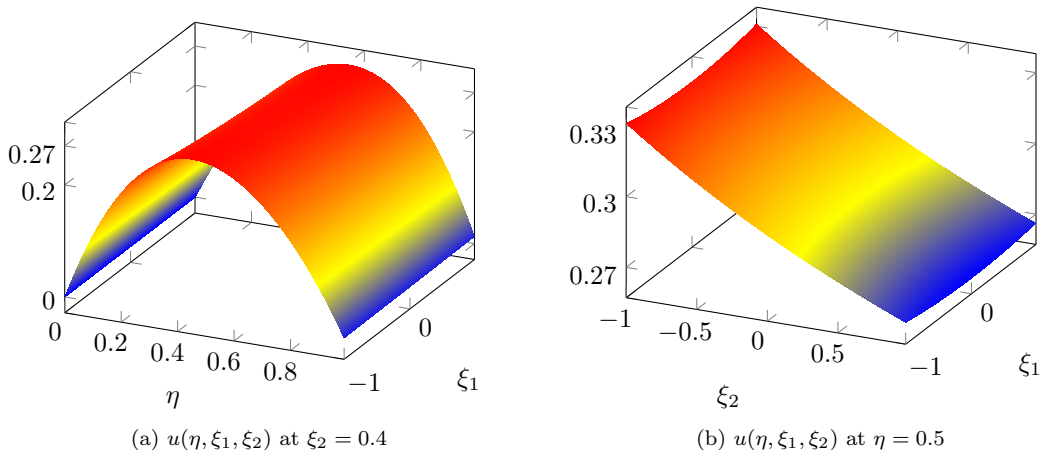
11

(a) $u(\eta, \xi_1, \xi_2)$ at $\xi_2 = 0.4$ (b) $u(\eta, \xi_1, \xi_2)$ at $\eta = 0.5$

FIG. 5.1. *Solution $u(\eta, \xi_1, \xi_2)$ of* (5.1) *for $d_{\boldsymbol{\xi}} = 2$ parameters and $a_0 = 4.3$, $\gamma = 2$.*

$$a(\eta, \boldsymbol{\xi}) = a_0 + \sum_{j=1}^{d_{\boldsymbol{\xi}}/2} \xi_{2j-1} \frac{\cos(j\pi\eta)}{j^\gamma} + \xi_{2j} \frac{\sin(j\pi\eta)}{j^\gamma},$$

with $\gamma \in \mathbb{R}$, $\gamma > 1$, $a_0 \in \mathbb{R}$, $a_0 > 2\zeta(\gamma)$, $d_{\boldsymbol{\xi}} \in 2\mathbb{N}$, and ζ denotes the Riemann zeta function. The random coefficient a is bounded in the interval $[a_0 - 2\zeta(\gamma), a_0 + 2\zeta(\gamma)]$ and thus the differential equation (5.1) is uniquely solvable for fixed $\boldsymbol{\xi} \in [-1, 1]^{d_{\boldsymbol{\xi}}}$.

The parameters ξ_k, $k = 1, \ldots, d_{\boldsymbol{\xi}}$ can be interpreted as random variables. Here we choose them to be uniformly distributed

$$\xi_k \sim U([-1, 1])$$

and we fix $a_0 = 4.3$ and $\gamma = 2$. In Figure 5.1 we (partially) plotted an approximation of the solution of this differential equation, where we restricted the number of random variables to two.

In order to demonstrate the applicability of the presented approach, we specify the settings of the applied algorithmic components. On the one hand, we restrict the numerical tests to the tent transform as periodization mapping, cf. Section 2.2.1, since this seems to be the most unfavourable choice due to its relatively low smoothness. On the other hand, we have to specify the applied sparse FFT approaches and the corresponding parameters. We choose three different sparsity levels s and refinements N for our approximated solutions $\breve{u}^\rho$, $\rho = \mathrm{I}, \mathrm{II}, \mathrm{III}$, cf. Section 3.2. Furthermore, we apply two different algorithms for computing approximate solutions denoted by $\breve{u}^\rho_{\mathrm{r1l}}$ and $\breve{u}^\rho_{\mathrm{mr1l}}$ namely the sFFT-algorithms that use sampling schemes that are *rank-1 lattices* and *multiple rank-1 lattices*, respectively. We call the corresponding sFFT algorithms R1LsFFT and MR1LsFFT. The basic structure of both algorithms is described in [18, Alg. 1]. The crucial differences of the R1LsFFT and the MR1LsFFT are in step 2b and 2f, where the first approach uses the component–by–component construction as described in "Algorithm 1" in [18, Sec. 2.2.1] in order to determine suitable generating vectors and the latter approach uses [13, Alg. 4] with $c = 2$ in order to determine multiple rank-1 lattice discretizations.

12

ρ	N	s	θ	r	$M_{\tilde{S}_{I_1}[\check{v}_1]}$	$M_{\tilde{S}_{I_2}[\tilde{v}_2]}$	$M_{S_{I_4}[w_1]}$	$|\hat{G}_N^{21}|$
I	32	1,000	$1\cdot10^{-12}$	5	$2.71\cdot10^7$	$2.96\cdot10^7$	$7.29\cdot10^6$	$1.18\cdot10^{38}$
II	64	5,000	$1\cdot10^{-12}$	5	$5.67\cdot10^8$	$7.15\cdot10^8$	$1.72\cdot10^8$	$2.1\cdot10^{44}$
III	128	8,000	$1\cdot10^{-12}$	5	$2.46\cdot10^9$	$3.06\cdot10^9$	$6.98\cdot10^8$	$4.06\cdot10^{50}$

TABLE 5.1

Parameter settings, the number of samples M used for the computation of the approximations of $\check{\tilde{v}}_1$, $\tilde{v}_2$, and $S_{I_4}[w_1]$ for $d_{\boldsymbol{\xi}} = 20$, cf. Sections 3.2 and 4, and the total cardinality of the box of frequency candidates for the sFFT algorithm that uses rank-1 lattices as spatial discretizations for $d_{\boldsymbol{\xi}} = 20$.

ρ	N	s	θ	r	$M_{\tilde{S}_{I_1}[\check{v}_1]}$	$M_{\tilde{S}_{I_2}[\tilde{v}_2]}$	$M_{S_{I_4}[w_1]}$	$|\hat{G}_N^{21}|$
I	32	1,000	$1\cdot10^{-12}$	5	$1.1\cdot10^8$	$1.15\cdot10^8$	$2.55\cdot10^7$	$1.18\cdot10^{38}$
II	64	5,000	$1\cdot10^{-12}$	5	$1.35\cdot10^9$	$1.45\cdot10^9$	$3.54\cdot10^8$	$2.1\cdot10^{44}$
III	128	8,000	$1\cdot10^{-12}$	5	$4.5\cdot10^9$	$4.75\cdot10^9$	$8.49\cdot10^8$	$4.06\cdot10^{50}$

TABLE 5.2

Parameter settings, the number of samples M used for the computation of the approximations of $\check{\tilde{v}}_1$, $\tilde{v}_2$, and $S_{I_4}[w_1]$ for $d_{\boldsymbol{\xi}} = 20$, cf. Sections 3.2 and 4, and the total cardinality of the box of frequency candidates for the sFFT algorithm that uses multiple rank-1 lattices as spatial discretizations for $d_{\boldsymbol{\xi}} = 20$.

Tables 5.1 and 5.2 show the parameters we used in columns two to five for the two different sFFT algorithms. The impacts of these parameters are shortly described in Section 2.1.

Increasing the number $d_{\boldsymbol{\xi}}$ of random variables yields approximation problems of higher dimensionality. Clearly for practical applications, the number of random variables needs to be suitably bounded. The used diffusion coefficient a is build in such a way, that the influence of the random variable ξ_j decreases with growing index j. Our first crucial task is to estimate the index j for which we can truncate the series expansion of a without losing significant information of a. In other words we would like to estimate a suitable number $d_{\boldsymbol{\xi}}$.

Example 5.1. To this end, we computed the approximation of a solution of (5.1) by our approach with a fixed large number $d_{\boldsymbol{\xi}} = 40$ of random variables, i.e., we treat an 41-dimensional approximation problem. We end up with an approximation $\check{u}_{\mathrm{r1l}}^{\mathrm{III}}$ as represented in (3.12). In order to simplify the considerations on the influence and the interactions on the variables of $\check{u}$ we apply periodizations and the sparse FFT approach on $\check{u}$ which leads in essence to a single Fourier sum representation of $\check{u}$. The associated frequency set of this approximate solution – together with the absolute values of the occuring (Fourier) coefficients of this solution – allow for rating the random variables to their importance. In particular, if the expansion $h_j - l_j$, $(k, \boldsymbol{h}), (k', \boldsymbol{l}) \in I$ of the frequency set in direction j is zero – or very small and the corresponding coefficients almost zero in relation to the largest occuring coefficients – the solution does not or not significantly depend on the variable ξ_j. Accordingly, leaving out this variable should not cause significant errors. Figure 5.2 indicates the expansions in each coordinate direction of the frequency set of $\check{u}$ for $d_{\boldsymbol{\xi}} = 40$. Obviously, the last 18 random variables have a very small expansion. We stress that the variables ξ_{21} as well as ξ_{22} have a significant frequency support but can be neglected due to the low order of magnitude of its Fourier coefficients. For these reasons, we restrict the number of random variables to $d_{\boldsymbol{\xi}} = 20$ in the following experiments.

As mentioned in the last example, we fix $d_{\boldsymbol{\xi}} = 20$. We solved (5.1) by the means of the sparse FFT approaches that uses single or multiple rank-1 lattices as spatial discretizations. The applied
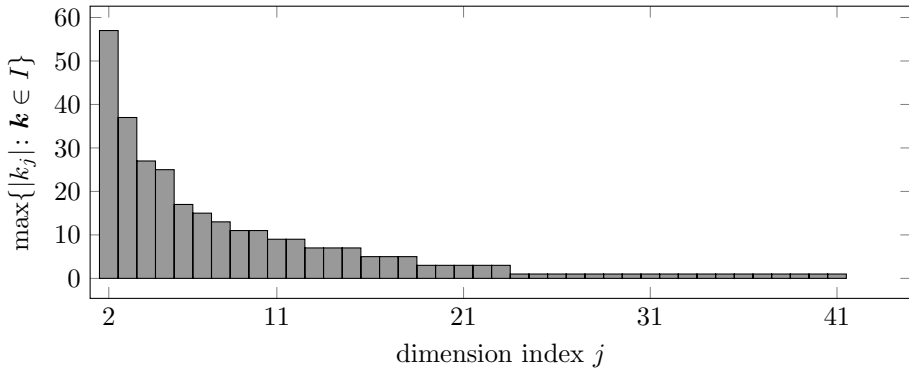
FIG. 5.2. *Directional expansion of the frequency set $I \subset \mathbb{Z}^{1+d_{\boldsymbol{\xi}}}$ of $\breve{u}$ for the random variables.*

parameter constellations are presented in Tables 5.1 and 5.2. Both tables contains the total amount of samples that were used for the approximation of the functions $\tilde{\tilde{v}}_1$, $\tilde{v}_2$, and $S_{I_4}[w_1]$ in columns six to eight for the different parameter settings as well. Moreover, the last columns of both tables present the cardinality of the full grids $\hat{G}_N^{21}$, where the sFFT algorithms search for the frequencies of the sparse representations of the computed approximations.

Example 5.2. We consider the average error of the computed approximations $\breve{u}_{\dagger}^{\rho}$, $\dagger \in \{\mathrm{r1l}, \mathrm{mr1l}\}$, of the solution u^* for fixed spatial nodes η_k. To this end we calculate the solution of (5.1) for $n_{\mathrm{test}} = 20000$ fixed randomly chosen parameters $\boldsymbol{\xi}^i \in [-1,1]^{20}$, $i = 1, \ldots, 20000$ as grid functions defined on the uniform grid

$$(5.2) \qquad\qquad \eta_k = \frac{k}{100}, \qquad k = 0, \ldots, 100,$$

via numerical integration and an error bound of 10^{-6}. We denote the corresponding solution by $\breve{u}$, i.e., we assume that the values $\breve{u}(\eta_k, \boldsymbol{\xi}^i)$ are suitable approximations of the true solution and we use these function values for comparison against our approximations. For a first comparison, we consider the pointwise difference with respect to our approximated solution and calculate the mean, i.e.

$$\mathrm{Err}_{\dagger}^{\rho}(\eta_k) := \frac{1}{n_{\mathrm{test}}} \sum_{i=1}^{n_{\mathrm{test}}} |\breve{u}(\eta_k, \boldsymbol{\xi}^i) - \breve{u}_{\dagger}^{\rho}(\eta_k, \boldsymbol{\xi}^i)|.$$

The errors $\mathrm{Err}_{\mathrm{r1l}}^{\rho}$ and $\mathrm{Err}_{\mathrm{mr1l}}^{\rho}$ for the parameter selections $\rho = \mathrm{I}, \mathrm{II}, \mathrm{III}$ from Tables 5.1 and 5.2 are plotted in Figures 5.3 and 5.4, respectively. We observe that the approximations computed by the MR1LsFFT are slightly better than those computed using the R1LsFFT algorithm. Certainly, this observation seems reasonable due to the usage of different numbers of sampling values, cf. Tables 5.2 and 5.1.

According to the last example, we computed an complete approximate solution of (5.1). In Section 4 we explained how to compute moments of these approximate solution.

Example 5.3. We demonstrate the performance of our approximation strategy by a comparison of subsequently computed approximate moments of the solution u^*. The Monte-Carlo approximation of
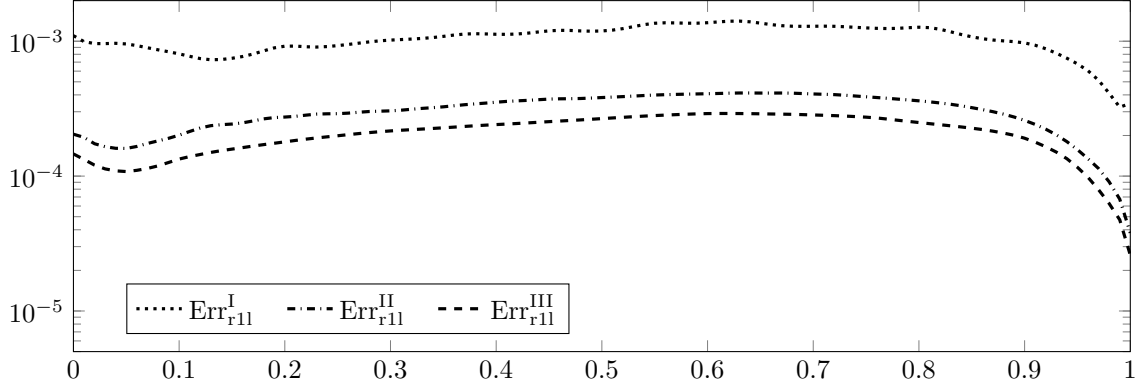
14

FIG. 5.3. *Averaged absolute errors* $\mathrm{Err}^\rho_{\mathrm{r1l}}$ *for 20000 random samples of y and using the sFFT algorithm with single rank-1 lattices sampling.*
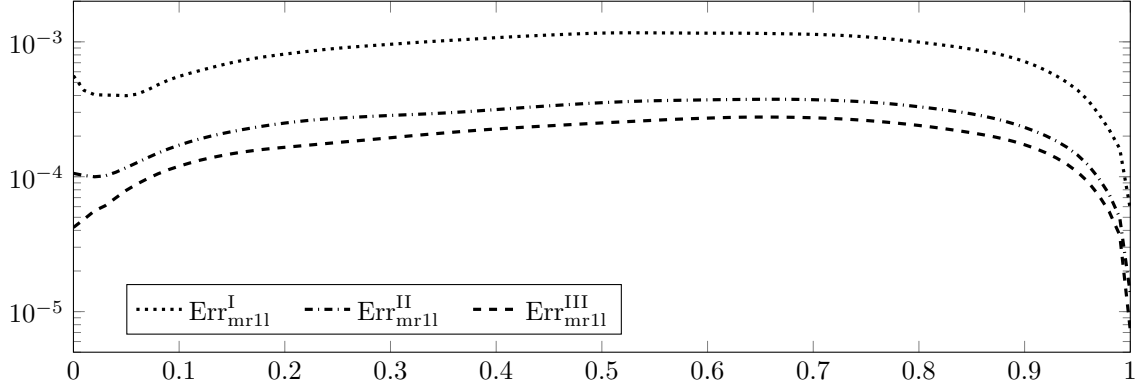

FIG. 5.4. *Averaged absolute errors* $\mathrm{Err}^\rho_{\mathrm{mr1l}}$ *for 20000 random samples of y and using the sFFT algorithm based on multiple rank-1 lattice sampling.*

the expectation value is given by

$$\overline{u_{n_{\mathrm{test}}}(\eta_k)} = \frac{1}{n_{\mathrm{test}}} \sum_{i=1}^{n_{\mathrm{test}}} \breve{u}(\eta_k, \boldsymbol{\xi}^i)$$

for fixed η_k, cf. (5.2), and the pointwise error at these spatial nodes is computed by

$$\mathrm{Res}^\rho_\dagger(\eta_k) := |\overline{u_{n_{\mathrm{test}}}(\eta_k)} - \mathbb{E}\breve{u}^\rho_\dagger(\eta_k)|,$$

where the approximations $\mathbb{E}\breve{u}^\rho_\dagger(\eta_k)$ of the first moment are gained from the solutions $\breve{u}^\rho_\dagger$, $\dagger \in \{\mathrm{r1l}, \mathrm{mr1l}\}$ as described in Section 4. The $\mathrm{Res}^\rho_\dagger$ behave very similar for fixed ρ and $\dagger \in \{\mathrm{r1l}, \mathrm{mr1l}\}$. Slightly better errors of the Expectation can be observed for the multiple rank-1 lattice approach, cf. Figures 5.5 and 5.6.

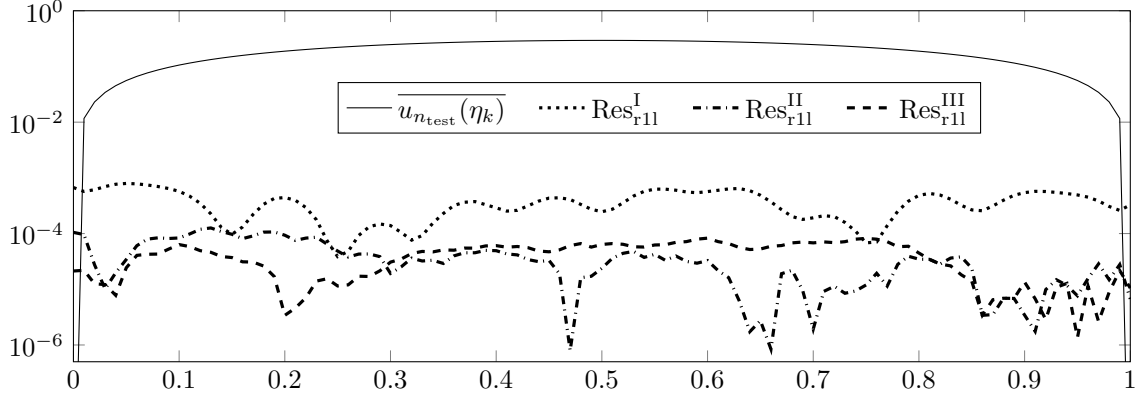Furthermore, we can regard higher order moments. In a similar way as above, we computed the

15

FIG. 5.5. *Absolute difference* $\mathrm{Res}^{\rho}_{\mathrm{r1l}}(\eta_k)$ *of Monte-Carlo expectation value* $\overline{u_{n_{\mathrm{test}}}(\eta_k)}$ *and the approximation* $\mathbb{E}\breve{u}^{\rho}_{\mathrm{r1l}}$ *of* u^*_{E}.
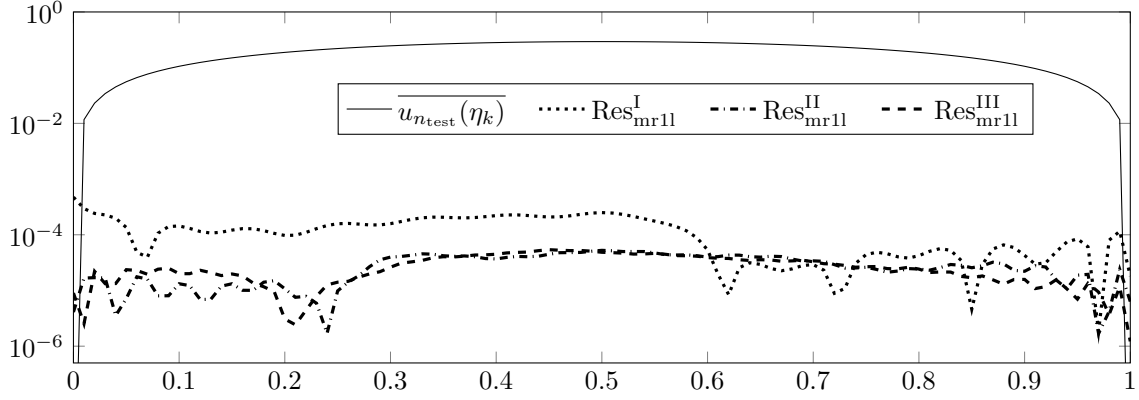

FIG. 5.6. *Absolute difference* $\mathrm{Res}^{\rho}_{\mathrm{mr1l}}(\eta_k)$ *of Monte-Carlo expectation value* $\overline{u_{n_{\mathrm{test}}}(\eta_k)}$ *and the approximation* $\mathbb{E}\breve{u}^{\rho}_{\mathrm{mr1l}}$ *of* u^*_{E}.

approximation of the second order moment by averaging

$$\overline{u_{n_{\mathrm{test}}}(\eta_k)^2} = \frac{1}{n_{\mathrm{test}}} \sum_{i=1}^{n_{\mathrm{test}}} \breve{u}(\eta_k, \boldsymbol{\xi}^i)^2$$

and the pointwise error for each η_k

$$\mathrm{Res}^{\rho,2}_{\dagger}(\eta_k) := |\overline{u_{n_{\mathrm{test}}}(\eta_k)^2} - \mathbb{E}\breve{u}^{\rho}_{\dagger}(\eta_k)^2|.$$

In Figure 5.7, the errors $\mathrm{Res}^{\rho,2}_{\mathrm{r1l}}$ for the single rank-1 lattice approach (R1LsFFT) are plotted. We see that adequately chosen parameters yield even very well approximated second moments.

16
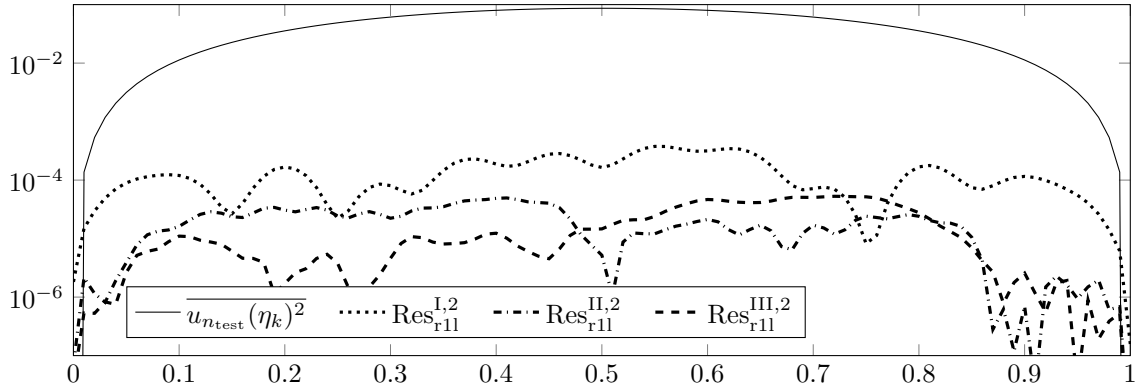
FIG. 5.7. *Absolute difference* $\mathrm{Res}^{\rho,2}_{\mathrm{r1l}}(\eta_k)$ *of Monte-Carlo second moment* $\overline{u_{n_{\mathrm{test}}}(\eta_k)^2}$ *and the approximation* $\mathbb{E}(\breve{u}^{\rho}_{\mathrm{r1l}})^2$ *of* $u^*_{\mathrm{E}^2}$.

References.

[1] M. Bachmayr, A. Cohen, R. DeVore, and G. Migliorati. Sparse polynomial approximation of parametric elliptic PDEs. part ii: lognormal coefficients. *ESAIM: M2AN*, 51:341–363, 2017.

[2] M. Bachmayr, A. Cohen, and G. Migliorati. Sparse polynomial approximation of parametric elliptic PDEs. part i: affine coefficients. *ESAIM: M2AN*, 51:321–339, 2017.

[3] J.-L. Bouchot, H. Rauhut, and C. Schwab. Multi-level Compressed Sensing Petrov-Galerkin discretization of high-dimensional parametric PDEs. *ArXiv e-prints*, Jan. 2017. arXiv:1701.01671 [math.NA].

[4] B. Choi, M. Iwen, and F. Krahmer. Sparse harmonic transforms: A new class of sublinear-time algorithms for learning functions of many variables. *ArXiv e-prints 1808.04932*, 2018.

[5] A. Cohen, R. DeVore, and C. Schwab. Analytic regularity and polynomial approximation of parametric and stochastic elliptic PDEs. *Anal. Appl. (Singap.)*, 9:11 – 47, 2010.

[6] R. Cools, F. Y. Kuo, D. Nuyens, and G. Suryanarayana. Tent-transformed lattice rules for integration and approximation of multivariate non-periodic functions. *J. Complexity*, 36:166–181, 2016.

[7] M. Eigel, M. Pfeffer, and R. Schneider. Adaptive stochastic Galerkin FEM with hierarchical tensor representations. *Preprint 2015-29, Preprint-Reihe des Instituts für Mathematik der TU Berlin*, 2015.

[8] I. Graham, F. Kuo, D. Nuyens, R. Scheichl, and I. Sloan. Quasi-Monte Carlo methods for elliptic PDEs with random coefficients and applications. *J. Comp. Phys.*, 230:3668 – 3694, 2011.

[9] A.-L. Haji-Ali, F. Nobile, L. Tamellini, and R. Tempone. Multi-index stochastic collocation for random PDEs. *Comput. Methods Appl. Mech. Engrg.*, 306:95–122, 2016.

[10] M. Hansen and C. Schwab. Analytic regularity and nonlinear approximation of a class of parametric semilinear elliptic PDEs. *Math. Nachr.*, 286(89):832–860, 2013.

[11] L. Kämmerer. *High Dimensional Fast Fourier Transform Based on Rank-1 Lattice Sampling*. Dissertation. Universitätsverlag Chemnitz, 2014.

[12] L. Kämmerer. Reconstructing multivariate trigonometric polynomials from samples along rank-1 lattices. In G. E. Fasshauer and L. L. Schumaker, editors, *Approximation Theory XIV: San Antonio 2013*, pages 255–271. Springer International Publishing, 2014.

[13] L. Kämmerer. Constructing spatial discretizations for sparse multivariate trigonometric polynomials that allow for a fast discrete Fourier transform. *Appl. Comput. Harmon. Anal.*, 2017, accepted.

[14] L. Kämmerer. Multiple rank-1 lattices as sampling schemes for multivariate trigonometric polynomials. *J. Fourier Anal. Appl.*, 24:17–44, 2018.

[15] L. Kämmerer, D. Potts, and T. Volkmer. High-dimensional sparse FFT based on sampling along multiple rank-1 lattices. *ArXiv e-prints 1711.05152*, 2017.

[16] O. P. Le Maître and O. M. Knio. *Spectral Methods for Uncertainty Quantification*. Scientific Computation.

Springer Netherlands, 2010.

[17] D. Potts and M. Tasche. Parameter estimation for multivariate exponential sums. *Electron. Trans. Numer. Anal.*, 40:204–224, 2013.

[18] D. Potts and T. Volkmer. Sparse high-dimensional FFT based on rank-1 lattice sampling. *Appl. Comput. Harmon. Anal.*, 41:713–748, 2016.

[19] H. Rauhut and C. Schwab. Compressive sensing Petrov–Galerkin approximation of high-dimensional parametric operator equations. *Math. Comp.*, 86:661–700, 2017.

[20] G. Suryanarayana, D. Nuyens, and R. Cools. Reconstruction and collocation of a class of non-periodic functions by sampling along tent-transformed rank-1 lattices. *J. Fourier Anal. Appl.*, 22:187–214, 2016.

[21] T. Volkmer. sparseFFTr1l, MATLAB® toolbox for computing the sparse fast Fourier transform based on reconstructing rank-1 lattices in a dimension incremental way. http://www.tu-chemnitz.de/~tovo/software, 2015.