Max-Planck-Institut
für Mathematik
in den Naturwissenschaften
Leipzig

Quantics-TT collocation approximation of parameter-dependent and stochastic elliptic PDEs

by

Boris N. Khoromskij, and Ivan V. Oseledets

Preprint no.: 37 2010

Quantics-TT collocation approximation of parameter-dependent and stochastic elliptic PDEs

Boris N. Khoromskij
Max-Planck-Institut für Mathematik in den Naturwissenschaften,
Inselstr. 22-26, D-04103 Leipzig, Germany; bokh@mis.mpg.de

Ivan Oseledets
INM RAS, Moscow, Russia; ivan@bach.inm.ras.ru

Abstract

We investigate the convergence rate of QTT stochastic collocation tensor approximations to solutions of multi-parametric elliptic PDEs, and construct efficient iterative methods for solving arising high-dimensional parameter-dependent algebraic systems of equations. Such PDEs arise, for example, in the parametric, deterministic reformulation of elliptic PDEs with random field inputs, based for example, on the M-term truncated Karhunen-Loève expansion. We consider both the case of additive and log-additive dependence on the multivariate parameter. The local-global versions of the QTT-rank estimates for the system matrix in terms of the parameter space dimension is proven. Similar rank bounds are observed in numerics for the solutions of the discrete linear system. We propose QTT-truncated iteration based on the construction of solution-adaptive preconditioner. Various numerical tests indicate that the numerical complexity scales almost linearly in the dimension of parametric space, and the adaptive preconditioner provides robust convergence in both additive and log-additive cases.

AMS Subject Classification: 65F30, 65F50, 65N35, 65F10
Key words: elliptic equations, stochastic PDEs, the Karhunen-Loève expansion, separable approximation, quantics-TT tensors, preconditioners, tensor-truncated iteration.

1 Introduction

In recent years tensor-structured numerical methods based on separation of variables have proved their value in multidimensional problems of computational chemistry [4, 29, 7, 14, 13], quantum molecular dynamics [21, 19], quantum computing [28], and stochastic PDEs [27, 26, 5, 10, 20, 16]. In particular, the low-rank rank tensor approximation in canonical format for stochastic PDEs were recently introduced in [16] (see also [18]).

In the present paper, we investigate the convergence rate of the quantics-TT (QTT) stochastic collocation tensor approximations for solving deterministic parametric elliptic equations in a high dimensional parameter space, arising, for example, as a projection of the stochastic PDE via a *truncated M-term Karhunen-Loève expansion*. In general, the model problem has the form

$$\mathcal{A}(y)u = f \quad \text{in } D \in \mathbb{R}^{d_0}, \tag{1.1}$$

$d_0 = 1, 2, 3$, where $\mathcal{A}(y)$ is an elliptic operator in a domain D, with coefficient depending on certain multidimensional parameter $y \in \mathbb{R}^M$, where M might very from several tens to several hundred. In the case of stochastic PDEs, we consider a class of model elliptic problems characterized by the additive/log-additive dependence of the equation coefficients on the multivariate parameter

y, corresponding to a random field that is linear/exponential in the random variable. Stochastic Galerkin approximation of equations (1.1) via sparse finite elements have been introduced in [2, 26], while stochastic-collocation method for elliptic sPDEs was described in detail in [1].

The QTT representation gives log-volume complexity in the number of elements of a tensor for several important problems [23, 12, 15]. For both additive and log-additive cases, we prove local and global versions of the QTT-rank estimates for the system matrix in terms of the parameter space dimension. We show that in some cases faster numerical algorithms can be designed using the localised version of QTT and canonical formats (cf. [13]). In particular, the local-QTT rank of system matrix is proven to be uniformly bounded in M, while its global rank is shown by numerical experiments to increase at most linearly in M. Uniform QTT rank bounds are observed in numerical tests for the solutions of respective discrete linear systems.

The principal idea of our approach is the iterative solution of a single coupled system of discrete, multiparametric elliptic equations projected onto the nonlinear manifold of low rank tensors represented in QTT format. The numerical cost of the matrix-vector multiplication in our setting scales linear in M, and in the discrete (physical) problem size N.

To enhance the convergence of the global solver, we propose the preconditioned block Jacobi-type iteration accomplished with the rank optimization at each iterative step. Our basic rank-1 preconditioner is constructed using the tensor-product approximation to the parametric elliptic operator inverse with the spatially homogeneous random coefficients. The overall numerical cost is estimated by $O(M^3 \log N) - O(M^4 \log N)$ provided that QTT rank of the system matrix is bounded by $O(M)$.

The rest of the paper is organized as follows. In §2, we set up the problem in the case of additive and log-additive coefficient dependence on the multivariate parameter. In §3, we first present the definitions of tensor structured vector- and matrix-formats to be utilized in the paper. We then discuss the tensor-product FEM-collocation scheme and prove the rank bounds for the resulting system matrix. We then describe the basic low tensor rank preconditioner, prove its spectral equivalence, and introduce the respective iterative solvers with adaptive rank optimization via QTT nonlinear approximation. In §4, we give various numerical examples corresponding to stochastic PDE with variable stochastic coefficients, in the case of random fields that are linear/exponential in the random variable. We investigate the case of both polynomial and exponential decay of stochastic coefficients. The numerical examples for elliptic equation with jumping parameter-dependent coefficients are also presented.

2 Parameter Dependent Elliptic Problem

2.1 Weak formulations and solvability

We consider parametric, elliptic problems which are posed in the physical domain $D := (0,1)^{d_0}$ of dimension $d_0 = 1, 2, 3$, and which depend on a vector of M parameters which take values in the hypercube in the M-dimensional parametric domain $\Gamma := (-1,1)^M \equiv I^M$, $M \in \mathbb{N}$. We are given $f \in L^2(D)$, and a parametric elliptic operator

$$\mathcal{A}(y) := -\mathrm{div}_x \left(a(y,x) \mathrm{grad}_x \right), \quad y \in \Gamma,$$

where the coefficient $a(y,x) = a_M(y,x)$ is a smooth function of $x \in D$ and the parameter vector $y = (y_1, ..., y_M) \in \Gamma$ with a possibly very large number M of parameters. We formulate the problems in the tensor-product Hilbert space (cf. [16]),

$$V := V_y \otimes V_x \quad \text{with } V_y := L^2(\Gamma) = \bigotimes_{m=1}^{M} L^2(I), \quad V_x := H_0^1(D).$$

Specifically, we are interested in the efficient numerical solution of the parametric elliptic problem: for every $y \in \Gamma$, find $u_M \in V$, such that

$$\mathcal{A}u_M(y,x) = f(x) \text{ in } D, \qquad u_M(y,x) = 0 \text{ on } \partial D. \tag{2.1}$$

In this problem setting the dimension M of the parametric space corresponds to the truncation parameter in the Karhunen-Loève expansion. In discretization of diffusion problems with random inputs, the dimension M of the parameter space could become arbitrarily large.

We consider the class of problems, with the *additive and log-additive* dependence of coefficient function on $y \in \Gamma$.

In the *additive case* the coefficient function is defined by

$$a_M(y,x) := a_0(x) + a_y(y,x), \quad \text{where } a_y(y,x) = \sum_{m=1}^{M} a_m(x) y_m, \tag{2.2}$$

with $a_m \in L^\infty(D),\ m = 1, ..., M$. Concerning the coefficient function $a_M(y,x)$, we assume (see [16]) that there exists $a_{min} > 0$, such that

1. $a_{min} \le a_0(x) < \infty$,
2. $\left| \sum_{m=1}^{M} a_m(x) y_m \right| < \gamma a_{min}$ with $\gamma < 1$, and for $|y_m| < 1$ $(m = 1, ..., M)$.

Conditions 1) - 2) imply the strong ellipticity of the problem (2.1) uniformly in y, i.e.,

$$a_M(y,x) \ge (1-\gamma) a_{min} > 0. \tag{2.3}$$

Hence, for $y \in \Gamma$, one can introduce the associated parametric bilinear form in the physical space V_x,

$$A(u,v) := \langle \mathcal{A}u, v \rangle_{L^2(D)} = \int_D a_M(y,x) \nabla_x u \cdot \nabla_x v dx \quad \forall u, v \in V_x,$$

so that we can use the respective to (2.2) additive splitting

$$A(u,v) = A_0(u,v) + A_y(u,v) \quad \forall u, v \in V_x,$$

where A_0 does not depend on $y \in \Gamma$. Under assumptions 1) - 2), we have the unique solvability for the corresponding weak formulation: for any $f \in H^{-1}(D)$ and for any $y \in \Gamma$, there exists a unique solution $u_M(y,\cdot) \in H_0^1(D)$ of the problem: Find $u_M \in V_x$, such that

$$\text{Find } u_M \in V_x, \text{ such that} \qquad A(u_M, v) = \int_D f(x) v(x) dx \quad \forall\ v \in V_x. \tag{2.4}$$

In elliptic problems the coefficient $a(x,y)$ should be positive which is not automatically satisfied by the affine mode (2.2), and it can be more natural to have operator coefficient a in form (the so-called *log-additive case*),

$$a(y,x) = e^{a_M(y,x)} = e^{a_0(x)} \prod_{m=1}^{M} e^{a_m(x) y_m}.$$

Conditions 1) - 2) imply spectral equivalence relations in physical variables,

$$C_0 \langle A_0 u, u \rangle \le \langle Au, u \rangle \le C_1 \langle A_0 u, u \rangle, \quad \forall\ u \in V_x, \tag{2.5}$$

with $C_0, C_1 > 0$, uniformly for all $y \in \Gamma$. Here A_0 is an elliptic operator corresponding to coefficient $a_0 = e^{a_0(x)}$, and A is an elliptic operator corresponding to coefficient $a = e^{a_M(y,x)}$. Hence, weak formulation (2.4) again has the unique solution. Notice that in the log-additive case the solvability conditions 1) - 2) can be essentially relaxed, but this issue is beyond the scope of this paper.

2.2 Stochastic-Galerkin and stochastic-collocation discretizations

The *parametric weak equation* (2.4) can be reformulated as the variational equation in the tensor-product Hilbert space V. Introducing the respective bilinear form

$$A_M(u,v) := \int_\Gamma \int_D a_M(y,x)\nabla_x u \cdot \nabla_x v dx dy \quad \forall\, u, v \in V,$$

we arrive at the following variational problem : Find $u_M \in V$, such that

$$A_M(u_M, v) = \int_\Gamma \int_D f(x)v(y,x)dxdy =: b_M(v) \quad \forall\, v \in V. \tag{2.6}$$

Lemma 2.1 *([12]) The equation (2.6) is uniquely solvable in V.*

Variational formulation (2.6) gives rise to the stochastic-Galerkin approximation of sPDEs.

Method proposed in this paper also applies to the stochastic-collocation approximation method. We refer to [1] on detailed description of stochastic-collocation method for elliptic sPDEs. We discretize the parametric equation (2.4) by Galerkin FEM or Finite Differences (FD) methods in physical domain D, and by collocation method in parameter domain Γ (see §3.3 for more detail).

3 Collocation discretization in parameter space

3.1 Formatted Tensor Representation of vectors and matrices

For the numerical solution of multiparameter problems low-parametric representations of high-dimensional arrays (tensors) that arise from numerical discretizations of such equation will be used. It will be shown, that all computations required to solve the multiparametric problem will be reduced to fast operations with high-dimensional structured matrices and vectors. These representations (called tensor formats) are crucial to avoid curse of dimensionality.

In this paper Tensor Train (TT) and (Quantics Tensor Train) (QTT) representation will be utilized heavily both for matrices and vectors (for detailed description see [22, 24, 23, 15, 12]), and for fast linear algebra operations in these formats TT-Toolbox[1] will be used.

The basic operations complexity is greatly reduced. For example, to multiply an $n^d \times n^d$ matrix with TT-ranks $r_k \leq r$, by a vector of length n^d with ranks $r'_k \leq r'$, the result is also a vector in TT-format with ranks bounded by rr', and the complexity is linear in the dimension d and polynomial in n and ranks r, r'. Basic facts, notations and algorithms for these format are summarized in the next section.

3.2 Notations and basic facts about TT and QTT formats

Basic objects used in this paper are multidimensional arrays, called tensors. They will be denoted by boldface letters, i.e. $\mathbf{A}$. An elements of a $n_1 \times n_2 \ldots \times n_d$ tensor $\mathbf{A}$ are denoted as $A(i_1, \ldots, i_d)$, and n_k are referred to as *mode sizes*. Since tensors belong to a linear space, standard linear operations (addition, multiplication by a number) are naturally defined. The Frobenius norm of a tensor, $||\mathbf{A}||_F$ is defined as

$$||\mathbf{A}||_F = \sqrt{\sum_{i_1,\ldots,i_d} A(i_1,\ldots,i_d)^2}.$$

[1] TT-Toolbox is publicly available from http://spring.inm.ras.ru/osel

An important operation is *tensor-by-matrix multiplication* over the mode-k (also called mode-product or contracted product). It is defined as

$$\mathbf{B} = \mathbf{A} \times_k U \quad \rightarrow \quad B(i_1, \ldots, i'_k, \ldots, i_d) = \sum_{i_k=1}^{n_k} A(i_1, \ldots, i_d) U(i_k, i'_k).$$

Tensors can be transformed into matrices in various ways. We adopt the following notation. Given a tensor $\mathbf{A} = A(i_1, i_2, \ldots, i_d)$ by

$$A(i_1 i_2 \ldots i_k; i_{k+1} \ldots i_d)$$

k-th unfolding matrix is denoted, i.e. first k indices enumerate its rows, and last $d-k$ — its columns.

A detailed review of tensors and their application can be found in [17, 13].

A multidimensional array (tensor) $\mathbf{A} = [A(i_1, \ldots, i_d)]$, $(1 \le i_k \le n_k)$ is said to be in the TT (tensor train) format, if it is represented as

$$A(i_1, i_2, \ldots, i_d) = \sum_{\alpha_1, \ldots, \alpha_{d-1}} G_1(i_1, \alpha_1) G_2(\alpha_1, i_2, \alpha_2) \ldots G_d(\alpha_{d-1}, i_d), \tag{3.1}$$

where α_k varies from 1 to r_k, and G_k are called *cores* of the TT-decompositions, and r_k are called compression ranks, or simply TT-ranks of the decomposition. Two border cores of the TT-format are matrices. To make decomposition more symmetric, it is natural to consider *extended form*:

$$A(i_1, i_2, \ldots, i_d) = \sum_{\alpha_0, \alpha_1, \ldots, \alpha_{d-1}, \alpha_d} G_1(\alpha_0, i_1, \alpha_1) G_2(\alpha_1, i_2, \alpha_2) \ldots G_d(\alpha_{d-1}, i_d, \alpha_d), \tag{3.2}$$

where two dummy indices α_0 and α_d are equal to one. This form simplifies the description of algorithms and will be used later on (corresponding ranks $r_0 = r_d = 1$).

Also, the k-th rank of the TT-decomposition of $\mathbf{A}$ will be denoted by $r_k(\mathbf{A})$. The TT-ranks are bounded from below by ranks of unfolding matrices A_k,

$$rank(\mathbf{A}) \ge r_k \ge \operatorname{rank} A_k, \quad k = 1, ..., d.$$

The unfolding procedure is also called matrization of a tensor [3]. The inverse statement is also true: if $r_k = \operatorname{rank} A_k$, then there exists a TT-decomposition with these ranks [22], and moreover it can be computed by d singular value decompositions (SVD) of auxiliary matrices [23]. Moreover, such procedure is stable: if unfolding matrices are of approximate low-rank r_k:

$$A_k = R_k + E_k,$$

where $\operatorname{rank} R_k = r_k$ and $||E_k||_F = \varepsilon_k$, then TT-approximate $\mathbf{B}$, computed by a sequence of SVD decompositions, satisfies

$$\|\mathbf{A} - \mathbf{B}\|_F \le \sqrt{\sum_{k=1}^{d-1} \varepsilon_k^2},$$

which confirms the stability of the approximation procedure (called TT-SVD later on, since it can be considered as a generalization of SVD algorithm for matrices).

If all ranks are equal to r and all mode dimensions are equal to n, then TT-format requires $\mathcal{O}(dnr^2)$ memory cells. Hence, the storage is linear in d and quadratic in r. The standard format to represent a d-dimensional array is the *canonical format*:

$$A(i_1, \ldots, i_d) \approx \sum_{\alpha=1}^{r} U_1(i_1, \alpha) \ldots U_d(i_d, \alpha). \tag{3.3}$$

It requires $\mathcal{O}(dnr)$ memory cells, however it suffers from certain drawbacks. Despite of recent progress, there are no robust algorithms to compute canonical decomposition numerically, and the approximation by a canonical tensor with a fixed rank can be ill-posed [6]. In contrast, computation of the best TT-approximation is a well-posed problem, and quasioptimal approximation can be computed by means of TT-SVD algorithm which uses standard LAPACK procedures, that is why it is more preferable in numerical computations.

One of the most important procedures in structured tensor computation is the *recompression procedure*. Given a tensor $\mathbf{A}$ in TT-format with non-optimal ranks r_k, we want to approximate in with another TT-tensor $\mathbf{B}$ with smallest possible ranks $\widehat{r}_k \leq r_k$ while maintaining the desired relative accuracy ε:

$$||\mathbf{A} - \mathbf{B}||_F \leq \varepsilon ||\mathbf{B}||_F.$$

Such projection will be denoted as

$$\mathbf{B} = T_\varepsilon(A).$$

Construction of such operator in the canonical form is a notoriously difficult task, with no best solution known.

For the TT-format it can be implemented by using standard algorithm from linear algebra (SVD and QR decompositions). Such algorithm is presented in [22]. For the completeness of the presentation, we give it here (in a notation slightly different from [22]).

A MATLAB code for this algorithm is a part of TT-Toolbox. By SVD_δ in Algorithm 1, we denote SVD with singular values that are set to zero if smaller then δ, and by $\mathrm{QR}_{\mathrm{rows}}$, we denote QR-decomposition of a matrix, where Q factor has orthonormal rows. The $\mathrm{SVD}_\delta(A)$ returns three matrices U, Λ, V of the decomposition $A \approx U\Lambda V^\top$ (as MATLAB **svd** function), and $\mathrm{QR}_{\mathrm{rows}}$ returns two: Q-factor and R-factor.

The complexity of the algorithm is $\mathcal{O}(dnr^3)$. All basic multilinear algebra (MLA) operations can be implemented in TT-format: addition, multiplication by a number, scalar product, norm, matrix-by-vector product. Together with the recompression procedure, this gives a nice tool for solving high-dimensional problems.

The QTT is a TT-decomposition applied to a special kind of tensors, arising from discretization of certain function. It was considered for tensorization of matrices ([23]) and vectors [12]. The simplest case is one-dimensional. Consider function $f(x)$ of one variable, $x \in [a, b]$, and discretize it on a uniform grid with 2^d points:

$$v_k = f(x_k), \quad k = 1, \ldots, 2^d.$$

The corresponding vector can be reshaped into a $2 \times 2 \times \ldots \times 2$, d-dimensional tensor, to which TT-decomposition can be applied. It appears that for many functions ranks r_k are very small [12], which leads to $\mathcal{O}(\log n)$ storage complexity for a vector of length n. Such an idea of introducing virtual dimensions can be generalized to higher dimensions. For example, for bivariate functions and their discretizations on a tensor grid, the corresponding tensor elements have the form

$$A(i_1, i_2, \ldots, i_d, j_1, \ldots, j_d), \quad 0 \leq i_k, j_k \leq 1.$$

Algorithm 1 TT-recompression

Require: d-dimensional tensor $\mathbf{A}$ in the TT-format, required accuracy ε
Ensure: $\mathbf{B}$ in the TT-format with smallest possibles compression ranks $\widehat{r}_k$ such that

$$||\mathbf{A}-\mathbf{B}||_F \leq \varepsilon ||\mathbf{A}||_F, \quad i.e. \quad \mathbf{B} = T_\varepsilon(\mathbf{A}).$$

1: Let G_k, $k = 1, \ldots, d-1$ be cores of $\mathbf{A}$.
2: {Initialization}
Compute truncation parameter $\delta = \frac{\varepsilon}{\sqrt{d-1}}||\mathbf{A}||_F$.
3: {Right-to-left orthogonalization}
4: **for** $k = d$ to 2 step -1 **do**
5: $[G_k(\beta_{k-1}; i_k\beta_k), R(\alpha_{k-1}, \beta_{k-1})] := \mathrm{QR}_{\mathrm{rows}}(G_k(\alpha_{k-1}; i_k\beta_k))$.
6: $G_{k-1} := G_k \times_3 R$.
7: **end for**
8: {Compression of the orthogonalized representation}
9: **for** $k = 1$ to $d-1$ **do**
10: {Compute δ-truncated SVD}
$[G_k(\beta_{k-1}i_k; \gamma_k), \Lambda, V(\beta_k, \gamma_k)] := \mathrm{SVD}_\delta[G_k(\beta_{k-1}i_k; \beta_k)]$.
11: $G_{k+1} := G_{k+1} \times_1 (V\Lambda)^\top$.
12: **end for**
13: Return G_k, $k = 1, \ldots, d$ as cores of $\mathbf{B}$.

TT-ranks crucially depend on the permutation of indices. For two-dimensional case, it was found experimentally [12] (using TT-SVD algorithm) that permutation of indices in the following fashion,

$$B(i_1j_1, i_2j_2, \ldots, i_dj_d),$$

i.e. with indices i_k, j_k interleaved, is preferable. Some theoretical results on the approximation in QTT-format were established in [12, 9].

To summarize, QTT format for a $2^d \times 2^d \times \ldots \times 2^d$, f-dimensional arrays leads to $\mathcal{O}(dfr^2)$ storage, and if r is small, it is of logarithmic complexity in the total number of elements, $O(\log 2^{df})$, of the array (log-volume complexity).

Both TT and QTT formats can be used also for matrices, with cores G_k indexed by four indices $G_k(\alpha_{k-1}, i_k, j_k, \alpha_k)$ instead of three. For such representation it is easily to implement matrix-by-vector product, where both matrix and vector are in the TT (QTT) format.

Now, we will discuss how QTT representation will be used for the discretization of parameter-dependent elliptic problems. Special structure of matrices can also be incorporated into cores G_k. For example, for sparse matrices in some modes corresponding tensors G_k will be sparse, and for diagonal dependence corresponding cores will be diagonal as well.

3.3 QTT representation of matrices and solution vectors

For each point $y \in \Gamma$ in parameter space we have an elliptic boundary value problem

$$\mathcal{A}(y, x)u(y, x) = f(x), \quad x \in D \in \mathbb{R}^{d_0},$$

which is discretized by a suitable approximation with N degrees of freedom in physical variable x. This yields a parametric linear system

$$A(y)v(y) = f, \quad f \in \mathbb{R}^N, \; v(y) \in \mathbb{R}^N, \quad y \in \Gamma. \tag{3.4}$$

The discretization can be performed by any suitable linear scheme (Galerkin, finite differences, projected collocation etc.).

In the following, we apply Galerkin collocation method. For each fixed $y \in \Gamma$, we discretize the parametric equation (2.4) by Galerkin FEM via a set of piecewise linear hat functions $\{\phi_i\}$, $i = 1, ..., N$. The main problem is the dependence on parameters $y = (y_1, \ldots, y_M)$. For each $m, 1 \leq m \leq M$, a one-dimensional grid of collocation points is introduced $\Gamma_m = \{y_m^{(k)}\} \in [-1, 1]$, $k = 1, \ldots, n$. This is equivalent to the collocation method, applied to (3.4), and the problem is reduced to n^M linear systems

$$A(j_1, \ldots, j_M)u(j_1, \ldots, j_M) = f, \quad 1 \leq j_k \leq n,$$

which can be written as one large linear system

$$\mathbb{A}\mathbf{u} = \mathbf{f}, \tag{3.5}$$

where $\mathbb{A}$ is a $Nn^M \times Nn^M$ matrix, $\mathbf{u}$ and $\mathbf{f}$ are vectors of length Nn^M.

In the *additive case* a parameter dependent matrix takes the form

$$A(y) = A_0 + \sum_{m=1}^{M} A_m y_m, \quad y \in \Gamma_M := \Gamma_m^M,$$

where A_m are $N \times N$ matrices and N is the number of degrees of freedom of discretization in x. In this case, $\mathbb{A}$ can be represented in a tensor form

$$\mathbb{A} = A_0 \times I \times \ldots \times I + A_1 \times D_1 \times I \times \ldots \times I + \ldots + A_M \times I \times \ldots \times D_M, \tag{3.6}$$

where D_m, $m = 1, \ldots, M$, is $n \times n$ diagonal matrix with positions of collocation points $\{y_m^{(k)}\}$, $k = 1, \ldots, n$, on the diagonal, and right-hand side has tensor rank 1:

$$\mathbf{f} = f \times e \times \ldots \times e,$$

where e is a vector of all ones of length n.

The same approach can be used also for *log-additive case*. The resulting linear system also has the form (3.4) - (3.5), but the dependence on y is no longer affine and special techniques should be used for matrix approximation in M-dimensional parameter space.

In log-additive case, system (3.5) has similar form, but there will be no straightforward low tensor rank representation to the matrix $\mathbb{A}$ like (3.6). However, still good low rank approximations of the form

$$\mathbb{A} \approx \sum_{k=1}^{R} \bigotimes_{m=0}^{M} A_{mk},$$

where matrices $A_{mk} \in \mathbb{R}^{(M+1)\times n}$, will be precomputed and utilized for fast calculations.

It is natural to consider low-parametric tensor formats for the solution $\mathbf{u}$, which can be regarded as a $(M + 1)$-dimensional tensor. Several options are available. Numerical solution of high-dimensional sPDEs in canonical format was considered [16]. Preliminary application of hierarchical Tucker format was addressed in [18].

In this paper, we apply QTT format to represent high-dimensional tensors in parametric variables. This representation gives log-volume complexity in the number of elements of a tensor for several important problems. It has all basic linear algebra operations implemented in MATLAB, and it can be effectively used for solving equations of form (3.5) by structured iterations with QTT-truncations as described in Section 4.

For the additive case, rank-$(M+1)$ representation of the matrix $\mathbb{A}$ is available at no cost. For log-additive case, it is very difficult to obtain canonical approximation to $\mathbb{A}$. However, QTT approximation to $\mathbb{A}$ can be computed by a certain fast procedure with recompression at each step. This procedure will be described in next section.

We show that in some cases the enhanced numerical algorithms can be designed using the localised version of QTT and canonical formats.

3.4 Matrix approximation in log-additive case

Let us describe, how to compute low parametric representation for a matrix in the log-additive case for a model one-dimensional example. Suppose D is $[0,1]$ and Galerkin discretization in x gives

$$A(i,j,y) = \int_D b(y,x)\frac{\partial \phi_i}{\partial x}\frac{\partial \phi_j}{\partial x}dx, \quad y \in \Gamma, \quad D=[0,1]. \tag{3.7}$$

As basis functions, take standard piecewise-linear hat functions $\{\phi_i\}$, $i = 1, ..., N$, and also apply simple quadratures to integrals (3.7). This gives (up to a factor h^2) a three-diagonal matrix $A(y)$ with elements

$$\begin{aligned} A(i,i,y) &= \frac{1}{4}(b(y,x_{i-1}) + 2b(y,x_i) + b(y,x_{i+1})), \\ A(i,i-1,y) &= \frac{1}{2}(b(y,x_{i-1}) + b(y,x_i)), \\ A(i-1,i,y) &= A(i,i-1,y), \end{aligned}$$

for $i = 1, ..., N$, and $y \in \Gamma$. Recall that

$$b(y,x) = e^{a_M(y,x)} = e^{a_0(x)} \prod_{m=1}^{M} e^{a_m(x)y_m}.$$

Hence, it is easy to represent matrix $A(y)$, $y \in \Gamma$, in a low tensor rank format,

$$A(y) = D(y) + Z(y) + Z^\top(y),$$

where $D(y)$ is a diagonal of A, Z is the first subdiagonal. First, consider $D(y)$. It can be represented as

$$D(y) = \sum_{i=1}^{N} A(i,i,y) e_i e_i^\top = \frac{1}{4}(C_1(y) + 2C_2(y) + C_3(y)),$$

where $C_2(y)$ can be written in the form

$$C_2(y) = \sum_{i=1}^{N} e_i e_i^\top e^{a_0(x_i)} \prod_{m=1}^{M} e^{a_m(x_i)y_m}. \tag{3.8}$$

After taking $C_2(y)$ in collocation points $y \in \Gamma_M$, diagonal matrix of size $Nn^m \times Nn^m$ appears, and it is easy to see that each summand in (3.8) has tensor rank-1. Moreover, if QTT format is considered in variable y_m, then TT-ranks will be equal to 1, since it is an exponential function [12]. This proves the following result.

Lemma 3.1 *For quadrature discretization of 1d PDE in the log-additive case each element* $\mathbf{A}(i,j,y)$ *has canonical decomposition of rank not larger than* 3. *Same QTT-rank bound holds true.*

This gives rise to a new format (local low-rank approximation) for representing a tensor. This format can be used directly to represent a matrix $\mathbb{A}$: each non-zero entry of a matrix is represented as a low-rank M-dimensional tensor. The full QTT-rank can also be estimated as in the following Lemma 3.2.

As a consequence of Lemma 3.1, we conclude that $C_2(y)$ has rank N at most. Ranks of matrices C_1, C_3 are also bounded by N. For $Z(y) = \frac{1}{2}(C_1(y) + C_2(y))$ we have two quadrature points, so the rank bound is $2N$, the same holds for $Z^\top(y)$, and the total rank estimate is $(1+1+1+2+2)N = 7N$. This estimate uses the fact that any fixed matrix element $f(y) = A(i, j, y)$ considered as a M-dimensional tensor, has the canonical rank bounded by a small constant, and there are only $\mathcal{O}(N)$ nonzero elements. This leads to the following result.

Lemma 3.2 *For quadrature discretization of 1d PDE in the log-additive case their exists rank-R canonical approximation to the assembled matrix $\mathbb{A}$, with the rank estimate*

$$R \leq 7N, \tag{3.9}$$

uniformly in n and M.

The arguments of Lemmas 3.1 and 3.2 also apply to arbitrary space dimension in physical variable $x \in \mathbb{R}^{d_0}$.

Remark 3.3 *Similar arguments prove that*

$$rank(A(i, j, y)) \leq C3^{d_0} \quad and \quad rank(\mathbb{A}) \leq C3^{d_0} N,$$

with constant C independent of N, n and M. Consequently, the QTT ranks of respective local matrices scale as $O(3^{d_0})$.

For one-dimensional problem with small grids (of order hundreds) it already gives a nice representation. However, for two-dimensional physical dimension N grows quadratically in the one-dimensional grid size and estimate (3.9) gives very large values of ranks.

However, in practice ranks are usually much smaller, normally in the range $O(M)$, and the canonical representation with overestimated rank R can be compressed by canonical-to-QTT compression algorithm [23] to QTT-format. The algorithm is simply addition of rank-one terms comprising the decomposition one-by-one and compressing the result afterwards to avoid excessive rank increase. This simple scheme allows fast computation of low-rank optimized TT-approximation to the matrix $\mathbb{A}$. It is summarized in Algorithm 2.

Recently in a work [5] it was shown, that under some mild assumptions in the additive case mapping $u(y, x)$ for each fixed x is analytic function of y, and showed existence of small-degree polynomial approximation to it. As a consequence, here comes a canonical rank estimate $C|\log\varepsilon|^{M-1}$, which is nice for small M, but still grows exponentially in M. However, as our numerical experiments demonstrate, this estimate is too pessimistic. Actually, the rank bound for the solution appeares to be almost uniform in M.

4 Preconditioned QTT-truncated iteration

QTT format can be effectively used for solving equations of form (3.5) by structured iterations with QTT-truncations of form

$$\widetilde{\mathbf{u}}^{(m+1)} := \mathbf{u}^{(m)} - \omega\mathbb{B}_m\left(\mathbb{A}\mathbf{u}^{(m)} - \mathbf{f}\right), \quad \mathbf{u}^{(m+1)} = T_\varepsilon(\widetilde{\mathbf{u}}^{(m+1)}) \to \mathbf{u}, \tag{4.1}$$

Algorithm 2 Matrix approximation in the log-additive case

Require: Subroutine that computes any element of a matrix $A(i,j,y)$, $1 \le i,j \le N$ in the QTT-format in the stochastic variable $y = (y_1, \ldots, y_M)$, with 2^p points in each stochastic mode, truncation parameter ε.

Ensure: QTT-approximation to matrix $\mathbb{A}$

1: Let $\mathcal{S}$ be sparsity pattern of A.
2: $\mathbf{A} := 0, M := 0$
3: **for** $(i,j) \in \mathcal{S}$ **do**
4: $\mathbf{B} := A(i,j,y)$ in QTT-format with cores $B_2, \ldots, B_{Mp+1}$
5: {Concatenate tensors $\mathbf{A}$ and $\mathbf{B}$ }
6: $r_k = r_k(\mathbf{A}), \widehat{r}_k = r_k(\mathbf{B})$, $A(1:M,:) = \mathbf{A}, A(M+1,:) = \mathbf{B}$ by concatenating cores:
7: $A_1 := \begin{pmatrix} A_1 & 0_{M \times \widehat{r_1}} \\ 0_{1 \times r_1} & 1_{1 \times \widehat{r_1}} \end{pmatrix}$.
8: **for** $k = 2$ to $Mp+1$ **do**
9: $C_k := 0_{(r_{k-1}+\widehat{r}_{k-1}) \times m_k \times (r_k + \widehat{r}_k)}$.
10: $C_k(1:r_{k-1}, :, 1:r_k) := A_k$
11: $C_k(r_{k-1}+1 : \widehat{r}_{k-1}, :, r_k + 1 : \widehat{r}_k) := B_k$.
12: $A_k := C_k$.
13: **end for**
14: $\mathbf{A} := T_\varepsilon(\mathbf{A})$.
15: **end for**
16: {How to treat the result}
$\mathbf{A}$ is a nnz $\times 2 \times \ldots \times 2$ dimensional tensor with $(Mp+1)$ dimensions
— "sparse" representation of $\mathbb{A}$
First core of $\mathbf{A}$, A_1 is now a nnz $\times\, r_1$ matrix, where nnz is the number of nonzeros in the sparsity pattern of $\mathbb{A}$. Each column of A_1 corresponds to $N \times N$ sparse matrix, and this is the sparse representation of the first core of $\mathbb{A}$. Other cores of $\mathbb{A}$ are diagonal matrices formed from cores of $\mathbf{A}$ (for each core of size $r_{k-1} \times 2 \times r_k$ tensor of size $r_{k-1} \times 2 \times 2 \times r_k$ is formed).
17: {Complexity}
The complexity of the algorithm is $\mathcal{O}(NMpr^3)$.

where T_ε is the rank truncation operator preserving accuracy ε.

At each step, TT-ranks increase, and have to be reduced by the recompression procedure from [22]. The procedure will work, if the solution indeed can be approximated in the TT format, and we will verify this numerically for several examples later on.

In the additive case of stochastic PDEs, a good choice of a preconditioner is a rank-1 tensor

$$\mathbb{B}_0 = A_0^{-1} \times I \times \ldots \times I,$$

see [16], which is easy to incorporate into the TT-framework. The spectral equivalence for this preconditioner is proved in [16].

However, in the more general setting the efficient preconditioner has to be chosen adaptively to the current iterand living in parametric space, without any prior knowledge. At each iteration the correction equation has to be solved approximately:

$$\mathbb{A}\mathbf{c} \approx \mathbf{f} - \mathbb{A}\mathbf{u},$$

or in terms of (nonlinear) preconditioner $\mathbb{B}$,

$$\mathbf{c} = \mathbb{B}(\mathbf{f} - \mathbb{A}\mathbf{u}).$$

To implement $\mathbb{B}$ we propose to use *one-point preconditioner scheme*, i.e. precondition by parameter-independent $N \times N$ matrix $A(y^*)$ for some special choice of multiparameter y^*:

$$\mathbf{c} = A(y^*)^{-1}(\mathbf{f} - \mathbb{A}\mathbf{u}).$$

This solves exactly only one of our systems, and y^* should be selected adaptively. As a simple heuristics the following scheme is proposed. For the residue tensor, $v(x, y) = \mathbf{f} - \mathbb{A}\mathbf{u}$, $x \in \mathbb{R}^N$, $y \in \Gamma_M$, we can find an approximate maximal element, where the residual is small, using multidimensional generalization of the maxvol algorithm [8]. This procedure is fast and inexpensive for a low-rank TT-tensor (it has complexity $\mathcal{O}(Nr^2 + pMr^3)$, as a result, we have the position of the maximum, (x^*, y^*), and use $A(y^*)$. We notice, that more natural approach is to compute the norms of all residuals in x for each y, and then compute the maximum. This procedure is more robust, but it is more expensive computationally and surprisingly in our experiments it usually gave worse convergence. Notice that the preconditioner $\mathbb{B}_0$ corresponds to the choice $y^* = 0$ in terms of continuous variable y.

After y^* is found, then preconditioner is defined by

$$\mathbb{B} = A(y^*)^{-1} \times I \times \ldots \times I.$$

This is a nonlinear preconditioner, since y^* is determined adaptively at each step, so not every linear solver can be used: the only option beyond the Richardson iteration is the geometric version of GMRES. To apply $\mathbb{B}$ to a TT-vector, one has to solve r independent $N \times N$ linear systems, where r is the first TT-rank of the vector. For elliptic problems any suitable fast solver can be used. Our examples are two-dimensional, so fast direct solver for sparse matrices can be applied. By $\mathbf{Solve}(N, r)$ we denote complexity to solve linear system of form

$$A(y^*)V = F_j, \quad j = 1, ..., r.$$

Finally, the solution algorithm looks as follows (the numerical complexity of respective steps in Algorithm 3 is specified in the right column).

5 Numerical experiments

5.1 Matrix approximation

We present numerical results for the approximation of the full system matrix in QTT format. For additive case, ranks grow linearly in M. For log-additive case no results are known, and we present first numerics for the approximation of the full matrix, corresponding to certain log-additive model examples.

Consider 2D-dimensional SPDE in stratified media (i.e., with coefficient depending on 1D variable) in the two cases:

1. Polynomial decay: $a_m(x) = \frac{0.5}{(m+1)^2} \sin mx$, $x \in [-\pi, \pi]$, $m = 1, \ldots, M$.

2. Exponential decay: $a_m(x) = e^{-0.7m} \sin mx$, $x \in [-\pi, \pi]$, $m = 1, \ldots, M$.

Algorithm 3 Richardson iteration with 1-point preconditioner

Require: Right-hand side f in QTT-format (f is can be considered as a $N\times 2\times 2\times\ldots\times 2$ $(1+Mp)$-dimensional tensor), block-diagonal matrix $\mathbb{A} = \mathbb{A}(i, i', j_1, j_2, \ldots, j_{Mp})$, $i, i' = 1, \ldots, N, j_k = 1, 2$, $k = 1, \ldots, Mp$ in QTT-format, truncation parameter ε, number of iterations n_{it}.

Ensure: Approximate solution x in QTT-format of the equation $\mathbb{A}x = f$.

1: {Initialization} $x = 0$

2: **for** $k = 1$ to n_{it} **do**

3: {Compute residual}
$\mathbf{res} = T_\varepsilon(\mathbb{A}x - f)$ $\qquad \mathcal{O}(N(r_1(x)r_1(\mathbb{A}))^2 + \sum_{m=2}^{Mp} r_m^2(x)r_m^2(\mathbb{A}))$
res is represented in QTT format:
$\mathbf{res}(i, j_1, \ldots, j_{Mp}) = \sum_{\alpha_1,\ldots,\alpha_{Mp}} R_1(i, \alpha_1)R_2(\alpha_1, j_1, \alpha_2)\ldots R_{Mp+1}(\alpha_{Mp}, j_{Mp})$

4: {Maximal residue}
$[i^*, j_1^*, j_2^*, \ldots j_{Mp}^*] = \arg\max |\mathbf{res}(i, j_1, j_2, \ldots, j_{Mp})|$ $\qquad \mathcal{O}(Nr_1^2(\mathbf{res}) + \sum_{m=2}^{Mp} r_m^3(\mathbf{res}))$

5: {1-point preconditioner}
Compute $N \times N$ sparse matrix B as
$B = A(i, i', j_1^*, \ldots, j_{Mp}^*)$
by contracting $\mathbb{A}$ over stochastic modes $\qquad \mathcal{O}(Nr_1(\mathbb{A}) + \sum_{m=2}^{Mp} r_m^2(\mathbb{A}))$

6: {Apply precondioner}
Solve for $BR' = R_1$, where R_1 is the first core of **res** using any suitable solver in physical space. $\qquad \mathbf{Solve}(N, r_1(\mathbf{res}))$

7: Set first core of **res** to R'

8: $x := x + \mathbf{res}$

9: $x := T_\varepsilon(x)$ $\qquad \mathcal{O}(Nr_1^2(x) + \sum_{m=2}^{Mp} r_m^3(x))$

10: **end for**

The parametric space is discretized on a uniform mesh in $[-1, 1]$ with 2^p points in each spatial direction. For the experiments, $p = 8$ is taken.

Ranks are presented with different truncation parameters. Table 5.1 presents results for the log-additive case and polynomial decay of coefficients, and Table 5.2 — for exponential decay. The dependence from M is linear for polynomial decay, and seems to be much milder in the case of exponential decay, which is rather natural.

M	QTT-rank(10^{-7})	QTT-rank(10^{-3})
5	27	10
10	44	17
20	78	27
40	117	49

Table 5.1: Rank-dependence of the matrix in the QTT format, 2D SPDE, log-additive case, polynomial decay $N = 128$, $p = 8$

M	QTT-rank(10^{-7})	QTT-rank(10^{-3})
5	33	11
10	43	21
20	51	23
40	50	25

Table 5.2: Rank-dependence of the matrix in the QTT format, 2D SPDE, log-additive case, exponential decay $N = 128$, $p = 8$

Table 5.3 describes dependence from the accuracy for a fixed M. This confirms, that ranks are logarithmic in accuracy ε.

ε	QTT-rank(ε)
10^{-3}	25
10^{-4}	31
10^{-5}	38
10^{-6}	44
10^{-7}	50

Table 5.3: Dependence of the matrix QTT-rank on accuracy, 2D SPDE, log-additive case, exponential decay $N = 128$, $M = 40$, $p = 8$

Tables 5.1 - 5.3 confirm numerically that matrices for log-additive case have low maximal QTT-ranks, and this representation can be used for the solution. In what follows two-dimensional model examples will be considered (i.e., $d_0 = 2$), both for additive and log-additive cases, as well as two multi-parameter problems will be studied.

We use two different TT rank estimates for tensors: one characterising the overall storage needs and complexity, $\hat{r}_{TT}$, and another one serving for the QTT-rank disctribution, $\overline{r}_{QTT}$:

$$\hat{r}_{TT}(\mathbf{u}) = \sqrt{\frac{\sum n_i r_i r_{i+1}}{\sum n_i}}, \quad \overline{r}_{QTT}(\mathbf{u}) = \sqrt{\frac{1}{M}\sum r_i r_{i+1}}.$$

5.2 Additive case

As the first example, consider two-dimensional diffusion in stratified media with diffusion coefficient

$$a(x_1, x_2, y_2, \ldots, y_M) = 1 + \frac{1}{2}\sum_{m=2}^{M} \lambda_m \sin(m x_1) y_m, \tag{5.1}$$

where $\lambda_m = e^{-0.7m}$. The results are presented in Figure 5.1 - 5.3.

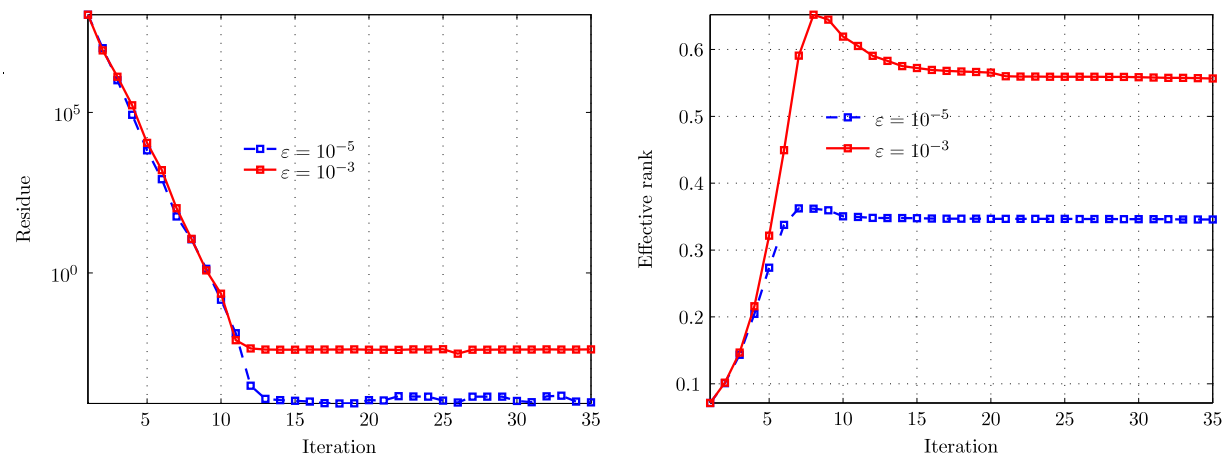

Figure 5.1: Convergence in the stratified 2D additive example with two different truncation parameters, 1-point preconditioner. Left: Residue with iteration; Right: $\hat{r}_{QTT}$-ranks with iteration, $M = 40$

Time dependence from M is presented in Figure 5.2

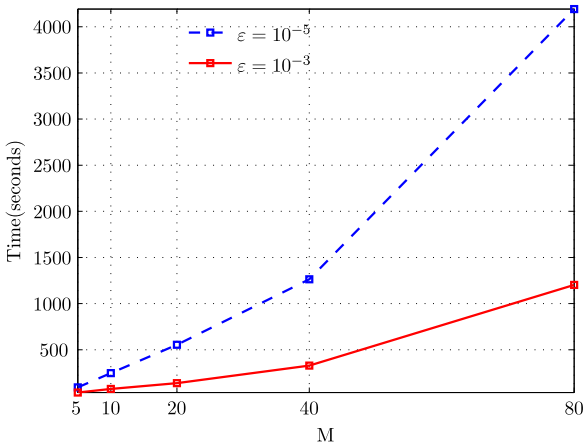

Figure 5.2: Time for 35 iterations, additive case, example (5.1)

Rank dependence from M is presented in Figure 5.3

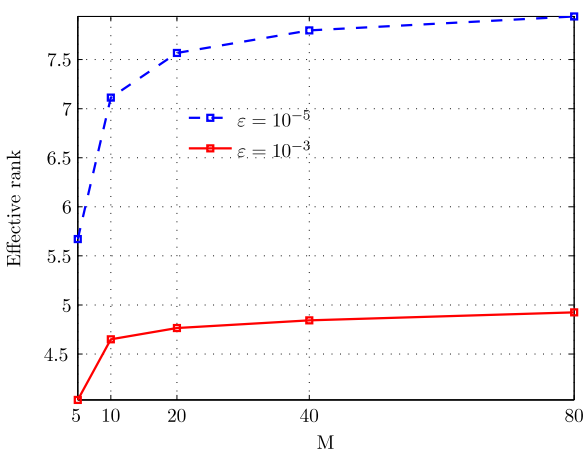

Figure 5.3: $\overline{r}_{QTT}$-ranks of the solution vs. M, additive case, example (5.1)

5.3 Log-additive case

As another example, consider two-dimensional diffusion in stratified media with diffusion coefficient

$$a(x_1, x_2, y_2, \ldots, y_M) = \exp(1 + \sum_{k=2}^{M} \lambda_k \sin(kx_1) y_k), \tag{5.2}$$

where $\lambda_k = e^{-0.7k}$. The results are presented in Figure 5.4.

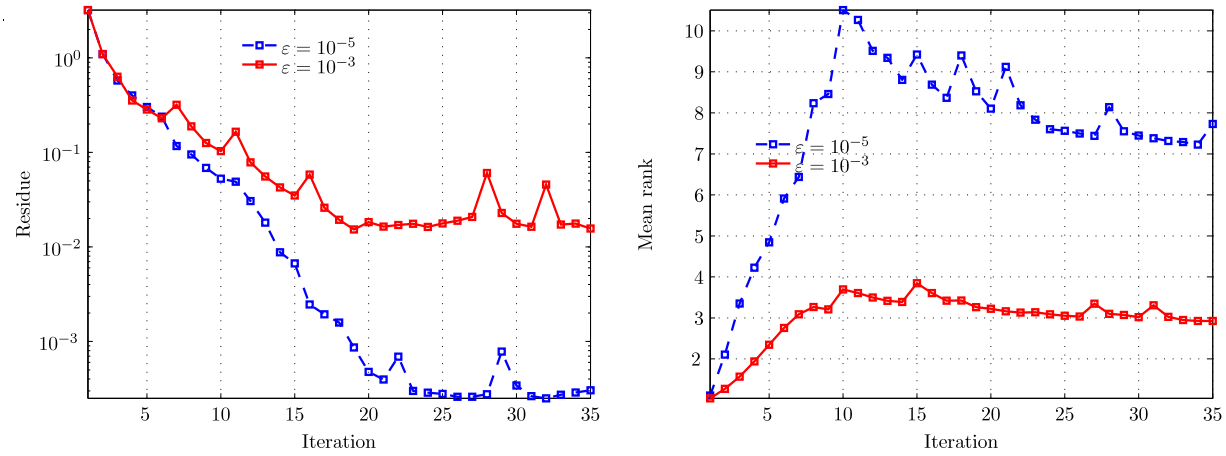

Figure 5.4: Convergence in the stratified 2D log-additive example with two different truncation parameters, 1-point preconditioner. Left: Residue with iteration; Right: $\overline{r}_{QTT}$-Ranks with iteration, $M = 40$.

Time dependence from M is presented in Figure 5.5

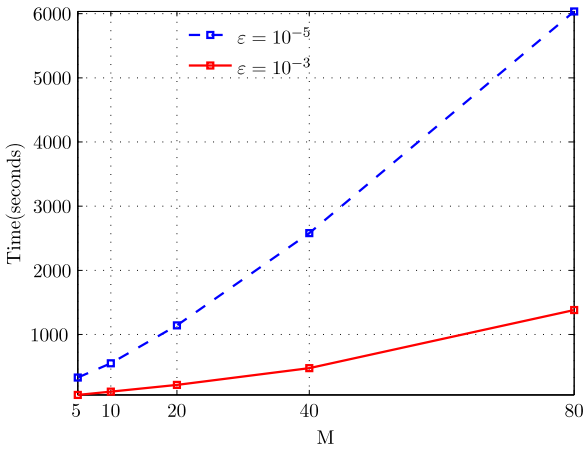

Figure 5.5: Time for 35 iterations, log-additive case, example (5.2)

5.4 Multi-parameter problems

In this subsection two "deterministic" problems with several parameters will be considered. The first one, borrowed from the work [2], is the following. Again, the diffusion equation

$$\mathrm{div}_x \left(a(y,x) \mathrm{grad}_x u \right) = f \in L^2(D), \quad y \in \Gamma,$$

considered in a square $D = [0,1]^2$, and the diffusion coefficient is represented as

$$a(x,y) = 1 + \sum_{i=1}^{4} \gamma_i \xi_i(x) y_i,$$

where $\xi_i(x)$ are indicator functions of four disks, see Figure 5.6, and $y_i \in [-0.99, 0]$, $i = 1, \ldots, 4$.

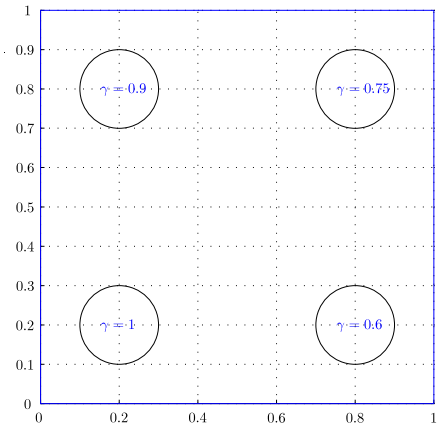

Figure 5.6: 4-circles test problem

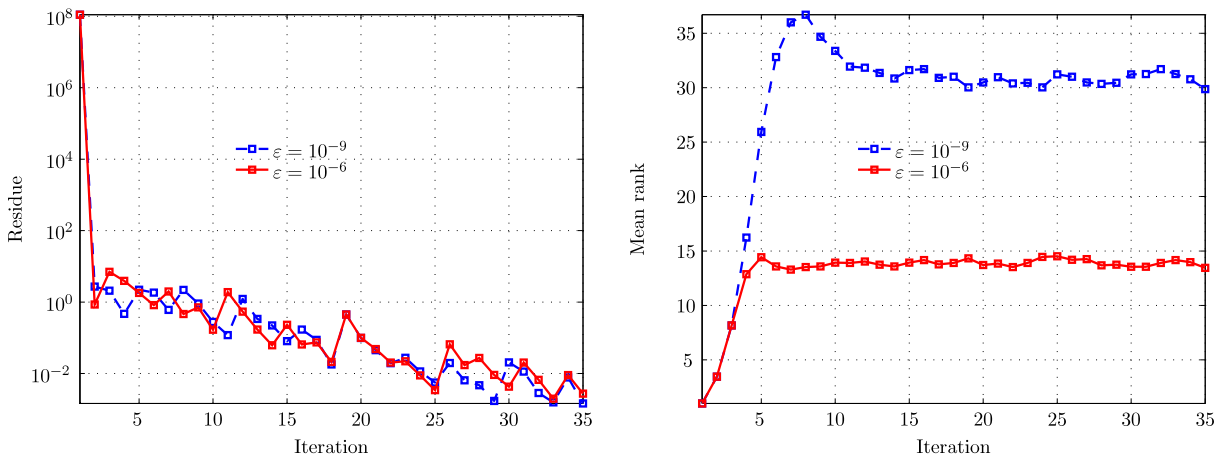

Figure 5.7: Convergence in the 4-circles example with two different truncation parameters, 1-point preconditioner, Left: Residue with iteration, Right: Ranks with iteration

Average time for one iterations was 0.22 seconds for $\varepsilon = 10^{-6}$ and 0.64 seconds for truncation parameter $\varepsilon = 10^{-9}$.

The second problem, is the stationary heat equation from Oberwolfach benchmarks (see [25]), considered in [18], which has form

$$A_0 u + (A_1 y_1 + A_2 y_2 + A_3 y_3) u = -b,$$

with $A_i, i = 1, \ldots, 4$ are 4257×4257 matrices, b is a vector of length 4257, and A_1, A_2, A_3, A_4 refer to different boundary conditions. This is related to boundary condition independent modelling [25] and such problems appear in compact thermal modelling ([11]). Parameters y_i vary from 10^8 to 10^9.

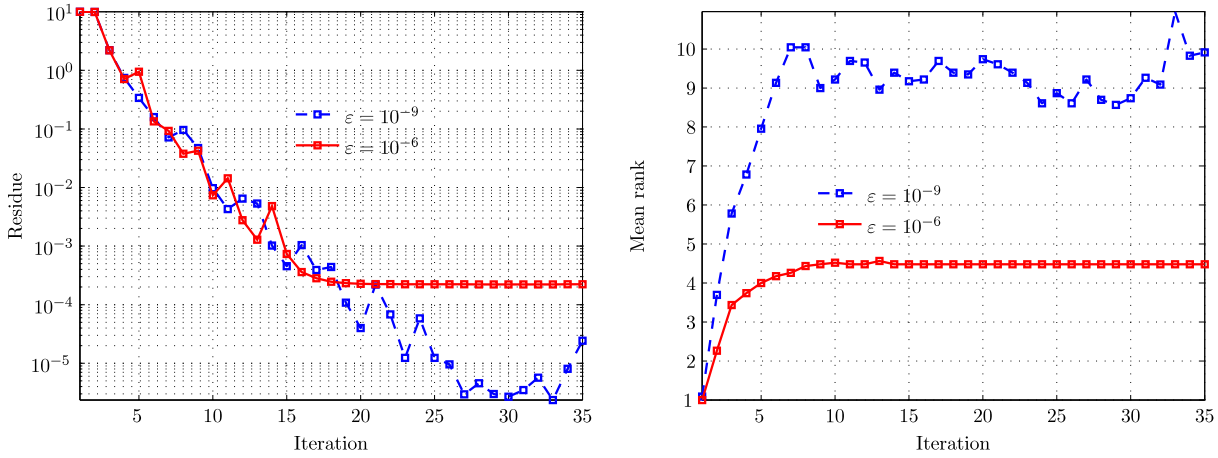

Figure 5.8: Convergence in the Oberwolfach example with two different truncation parameters, 1-point preconditioner, Left: Residue with iteration, Right: Ranks with iteration

Average time for one iterations was 0.22 seconds for $\varepsilon = 10^{-6}$ and 0.64 seconds for truncation

parameter $\varepsilon = 10^{-9}$.

6 Conclusions

We have presented first application of QTT format to the solution of high-dimensional equations, arising from stochastic PDE and parameter-dependent elliptic equations. It was proved that ranks in QTT matrix format in both the additive and log-additive cases are bounded by a constant, independent of M. However, it depends on N — physical problem size. The proof of the estimate is constructive and gives an approximation algorithm. Using this algorithm we showed by numerical experiments, that the estimate is rather pessimistic — actually ranks scale linearly in M in the worst case, and linearly in $\log \varepsilon$, where ε is the accuracy of approximation. Similar rank behaviour is observed for solution of the equation. To solve the equation we use QTT-truncated iteration with adaptive block-Jacobi-like preconditioner, which demonstrated linear convergence rate in our numerical experiments. To summarize, the proposed method looks promising for the approximation and solution of parameter-dependent equations in the case of additive and log-additive coefficient dependence on parameters.

References

[1] I. Babuska, F. Nobile, and R. Tempone, *A Stochastic Collocation Method for Elliptic Partial Differential Equations with Random Input Data*, SIAM Journal on Numerical Analysis, 45 (2007), pp. 1005–1034.

[2] I. Babuska, R. Tempone, and G. E. Zouraris, *Galerkin Finite Element Approximations of Stochastic Elliptic Partial Differential Equations*, SIAM Journal on Numerical Analysis, 42 (2004), pp. 800–825.

[3] B. W. Bader and T. G. Kolda, *Efficient MATLAB Computations with Sparse and Factored Tensors*, SIAM Journal on Scientific Computing, 30 (2008), p. 205.

[4] G. Beylkin and M. J. Mohlenkamp, *Algorithms for Numerical Analysis in High Dimensions*, SIAM Journal on Scientific Computing, 26 (2005), p. 2133.

[5] A. Cohen, R. DeVore, and C. Schwab, *Convergence rates of best N-term Galerkin approximations for a class of elliptic sPDEs*, 2009.

[6] V. de Silva and L.-H. Lim, *Tensor Rank and the Ill-Posedness of the Best Low-Rank Approximation Problem*, SIAM Journal on Matrix Analysis and Applications, 30 (2008), p. 1084.

[7] I. Gavrilyuk, W. Hackbusch, and B. Khoromskij, *Tensor-product approximation to elliptic and parabolic solution operators in higher dimensions*, Computing, 74 (2005), pp. 131–157.

[8] S. A. Goreinov, I. V. Oseledets, D. V. Savostyanov, E. E. Tyrtyshnikov, and N. L. Zamarashkin, *How to find a good submatrix*, in Matrix methods: theory, algorithms, applications, V. Olshevsky and E. Tyrtyshnikov, eds., NY, 2010, World Scientific Publishers, pp. 247–256.

[9] L. Grasedyck, *Polynomial Approximation in Hierarchical Tucker Format by Vector-Tensorization*, 2010.

[10] H. Harbrecht, R. Schneider, and C. Schwab, *Sparse second moment analysis for elliptic problems in stochastic domains*, Numerische Mathematik, 109 (2008), pp. 385–414.

[11] W. Huang, M. R. Stan, K. Skadron, K. Sankaranarayanan, S. Ghosh, and S. Velusam, *Compact thermal modeling for temperature-aware design*, Annual ACM IEEE Design Automation Conference, (2004).

[12] B. N. Khoromskij, *O(d logN)-Quantics Approximation of N-d Tensors in High-Dimensional Numerical Modeling*, 2009.

[13] ——, *Tensors-structured Numerical Methods in Scientific Computing: Survey on Recent Advances*, December 2010.

[14] B. N. Khoromskij, V. Khoromskaia, and H.-J. Flad, *Numerical solution of the Hartree-Fock equation in multilevel tensor-structured format*, 2009.

[15] B. N. Khoromskij and I. V. Oseledets, *Quantics-TT approximation of elliptic solution operators in higher dimensions*, 2010.

[16] B. N. Khoromskij and C. Schwab, *Tensor-structured Galerkin approximation of parametric and stochastic elliptic PDEs*, 2010.

[17] T. G. Kolda and B. W. Bader, *Tensor Decompositions and Applications*, SIAM Review, 51 (2009), p. 455.

[18] D. Kressner and C. Tobler, *Iterative solution of parameter-dependent linear Systems.*, 2010.

[19] C. Lubich, *From quantum to classical molecular dynamics: reduced models and numerical analysis*, EMS, Zurich, 2008.

[20] H. Matthies and A. Keese, *Galerkin methods for linear and nonlinear elliptic stochastic partial differential equations*, Computer Methods in Applied Mechanics and Engineering, 194 (2005), pp. 1295–1331.

[21] H.-D. Meyer, F. Gatti, and G. A. Worth, *Multidimensional Quantum Dynamics: MCTDH Theory and Applications*, Wiley-VCH, Weinheim, 2009.

[22] I. V. Oseledets, *Compact matrix form of the d-dimensional tensor decomposition*, 2009.

[23] ——, *Tensors inside of matrices give logarithmic complexity*, 2009.

[24] I. V. Oseledets and E. E. Tyrtyshnikov, *Breaking the Curse of Dimensionality, Or How to Use SVD in Many Dimensions*, SIAM Journal on Scientific Computing, 31 (2009), p. 3744.

[25] E. B. Rudnyi and J. G. Korvink, *Model Order Reduction of MEMS for Efficient Computer Aided Design and System Simulation*, 2008.

[26] C. Schwab and R. A. Todor, *Sparse finite elements for elliptic problems with stochastic loading*, Numerische Mathematik, 95 (2003), pp. 707–734.

[27] R. A. Todor and C. Schwab, *Convergence rates for sparse chaos approximations of elliptic problems with stochastic coefficients*, IMA Journal of Numerical Analysis, 27 (2006), pp. 232–261.

[28] G. Vidal, *Efficient Classical Simulation of Slightly Entangled Quantum Computations*, Physical Review Letters, 91 (2003).

[29] H. Yserentant, *Sparse grid spaces for the numerical solution of the electronic Schrödinger equation*, Numerische Mathematik, 101 (2005), pp. 381–389.