ESAIM: M2AN 50 (2016) 1107–1136
DOI: 10.1051/m2an/2015071

**ESAIM: Mathematical Modelling and Numerical Analysis**
www.esaim-m2an.org

# ADAPTIVE LOW-RANK METHODS FOR PROBLEMS ON SOBOLEV SPACES WITH ERROR CONTROL IN $L_2$ *

M. Bachmayr[1] and W. Dahmen[2]

**Abstract.** Low-rank tensor methods for the approximate solution of second-order elliptic partial differential equations in high dimensions have recently attracted significant attention. A critical issue is to rigorously bound the error of such approximations, not with respect to a fixed finite dimensional discrete background problem, but with respect to the exact solution of the continuous problem. While the energy norm offers a natural error measure corresponding to the underlying operator considered as an isomorphism from the energy space onto its dual, this norm requires a careful treatment in its interplay with the tensor structure of the problem. In this paper we build on our previous work on energy norm-convergent subspace-based tensor schemes contriving, however, a modified formulation which now enforces convergence only in $L_2$. In order to still be able to exploit the mapping properties of elliptic operators, a crucial ingredient of our approach is the development and analysis of a suitable asymmetric preconditioning scheme. We provide estimates for the computational complexity of the resulting method in terms of the solution error and study the practical performance of the scheme in numerical experiments. In both regards, we find that controlling solution errors in this weaker norm leads to substantial simplifications and to a reduction of the actual numerical work required for a certain error tolerance.

**Mathematics Subject Classification.** 41A46, 41A63, 65D99, 65J10, 65N12, 65N15.

Received December 12, 2014. Revised May 6, 2015. Accepted September 10, 2015.

## 1. Introduction

For a given open product domain $\Omega = \Omega_1 \times \ldots \times \Omega_d \subset \mathbb{R}^d$, we are interested in approximately solving problems of the form

$$-\operatorname{div}(M \operatorname{grad} u) = f \text{ in } \Omega, \quad u|_{\partial\Omega} = 0, \tag{1.1}$$

*Keywords and phrases.* low-rank tensor approximation, adaptive methods, high-dimensional elliptic problems, preconditioning, computational complexity.

* *This work has been supported in part by the DFG SFB-Transregio 40, the Excellence Initiative of the German Federal and State Governments (RWTH Aachen Distinguished Professorship), NSF grant DMS 1222390, and the ERC advanced grant BREAD.*

[1] Sorbonne Universités, UPMC Université Paris 06, CNRS, UMR 7598, Laboratoire Jacques-Louis Lions, 4 place Jussieu, 75005, Paris, France. bachmayr@ljll.math.upmc.fr

[2] Institut für Geometrie und Praktische Mathematik, RWTH Aachen, Templergraben 55, 52056 Aachen, Germany. dahmen@igpm.rwth-aachen.de

Article published by EDP Sciences

where $M$ is a symmetric uniformly positive definite $(d \times d)$-matrix over $\Omega$. Here, we are interested in the *spatially high-dimensional* regime $d \gg 1$. Specifically, our subsequent analysis is based on the assumption that $M$ is diagonally dominant with constant entries. However, concerning variable coefficients, all findings carry over to diagonal matrices $M = \operatorname{diag}(M_1, \dots, M_d)$, where the $M_i$ are sufficiently benign functions of $x_i$, $i = 1, \dots, d$. Due to the coupling of variables when $M$ is non-diagonal one expects that tensor approximations exhibit a stronger rank growth when accuracy tolerances decrease. In our numerical experiments we show that this effect can already be significant for constant tridiagonal matrices $M$.

For simplicity of exposition, we deliberately keep (1.1) on the level of a specific model problem. However, what follows applies in essence also to natural variants of (1.1), for instance, when the Dirichlet boundary conditions are replaced (partially or throughout) by Neumann conditions, as long as the type of boundary condition remains the same on each $(d-1)$-face of $\Omega$. While above the $\Omega_i$ are intervals, one could also consider a product of $d$ more general low-dimensional domains.

For product domains and weakly coupling diffusion matrices, the differential operator in (1.1) has formally low rank, that is, its action only leads to a moderate increase in the ranks of suitable tensor representations. This justifies the hope that, for instance, for separable right hand sides the solution may be approximable efficiently by low-rank tensor expansions, where the low-dimensional factors in the rank-one summands are not predefined basis functions but are allowed to depend on the solution. This hope is indeed supported by substantial numerical evidence [5, 13] and by rank bounds for approximate solutions of a structured discrete linear system obtained from a fixed discretization of the continuous problem [15, 25]. This does not, however, provide any information on how the ranks grow when the discretization is refined. At least for diagonal $M$, a rigorous quantitative interrelation between the accuracy of approximations to the continuous exact solution $u$ of (1.1) in appropriate function spaces and the required ranks is given in [11].

The numerical treatment of such solution-dependent basis functions still necessitates their expansion in terms of suitable low-dimensional *reference* basis functions. In this combination of low-rank representations and basis expansions of corresponding tensor components, we are thus in fact dealing with two levels of approximation. Between these, a proper balance needs to be maintained in the convergence to the exact solution, since both allowing large tensor ranks for coarse discretizations and using very fine discretizations with inaccurate low-rank approximations will, especially at high accuracies, lead to excessive numerical costs. Common strategies for low-rank approximations start from a fixed discretizations and use tensor representations as a linear algebra tool, see *e.g.* [5, 7, 24]. Such a fixed discretization corresponds to a fixed finite reference basis for representing the tensor components. In this setting, however, one cannot address the necessary intertwining of subspace approximation and adaptive refinement of tensor factor representations. We thus need to deal with several closely connected issues: obtaining *a posteriori* error information that can guide the adaptive refinement of the reference basis, ensuring that the underlying representation of the differential operator does not become ill-conditioned as the basis is refined, and avoiding inappropriately large tensor ranks.

These considerations have motivated the approach put forward in [2, 4] on a general level and in [3] with special focus on problems of the form (1.1). A central idea there is that rigorous *a posteriori* error bounds driving convergent approximation schemes should rely on a faithful approximation to the *residual of the continuous problem* which, in turn, should reflect the accuracy of the approximate solution. This seems to be possible only when exploiting the mapping properties of the operator $A$ induced by the classical weak formulation

$$\langle Au, v\rangle := \int_\Omega M \operatorname{grad} u \cdot \operatorname{grad} v \, \mathrm{d}x = \langle f, v\rangle, \quad v \in \mathrm{H}^1_0(\Omega), \tag{1.2}$$

over the space $\mathrm{H}^1_0(\Omega)$. In fact, denoting by $\underline{c}, \bar{C}$ the smallest and largest eigenvalue of $M$ one has

$$\|A\|_{\mathrm{H}^1_0(\Omega)\to\mathrm{H}^{-1}(\Omega)} \le \bar{C}, \quad \|A^{-1}\|_{H^{-1}(\Omega)\to\mathrm{H}^1_0(\Omega)} \le \underline{c}^{-1}, \tag{1.3}$$

which is equivalent to saying that errors in the $\mathrm{H}^1$-norm can be faithfully estimated by residuals in the dual norm $\|\cdot\|_{\mathrm{H}^{-1}(\Omega)}$.

It is unfortunately not entirely straightforward to exploit these facts for a rigorous error control of low-rank tensor approximations. Subspace-based tensor formats, whose stability properties play an important role in devising reliable computational routines, are not immediately amenable to spaces that are not endowed with cross-norms, see [3] for a detailed discussion. Therefore, the strategy in [2–4] is based on transforming the problem first to an equivalent one where the transformed operator $\mathbf{A}$ is an isomorphism mapping an $\ell_2$-space over an infinite product index set, which *is* a space endowed with a cross-norm, *onto itself*. This transformation requires a Riesz basis for the energy space $\mathrm{H}^1_0(\Omega)$. A suitable basis of this type can be obtained by rescaling an orthonormal tensor product wavelet basis of $\mathrm{L}_2(\Omega)$. Unfortunately, and this is the price to be paid, the rescaling destroys separability of the basis functions and, as a consequence, causes the resulting operator representation $\mathbf{A}$ to have *infinite rank*. Aside from the role of suitable recompression and coarsening operators given in [4], a key ingredient in still constructing low-rank approximations with controlled energy norm accuracy for elliptic problems are *adaptive finite-rank rescaling operators* proposed and analyzed in [3]. They are based on specially tailored relative error bounds for exponential sum approximations to the function $g(t) = t^{-1/2}$. This ultimately led to an adaptive refinement scheme generating approximate solutions represented in hierarchical tensor formats, convergent in energy norm with near-optimal complexity, for each fixed spatial dimension $d$, with respect to ranks and representation sparsity of the tensor factors [3].

Nevertheless, the fact that the energy norm is not a cross-norm and the resulting unbounded tensor ranks of the representation $\mathbf{A}$ significantly impede the control of rank growth in the iterates. The central question addressed in the present work is therefore: *Can one devise a solver that provides approximate solutions in hierarchical tensor format at a significantly lower numerical cost by enforcing convergence only in a norm that is* weaker *than the energy norm, namely* $\|\cdot\|_{\mathrm{L}_2(\Omega)}$*?*

Of course, there is no hope of avoiding the above mentioned "*scaling problem*" completely. In one way or another, a rigorous convergence analysis has to make use of the mapping properties of the underlying operator, which always refers to a pair of spaces of which at least one is *not* endowed with a cross-norm. However, if one has *full elliptic regularity*, the underlying operator is also an isomorphism from $\mathrm{H}^2(\Omega)\cap\mathrm{H}^1_0(\Omega)\to\mathrm{L}_2(\Omega)$ and, by duality, also from $\mathrm{L}_2(\Omega)$ onto $(\mathrm{H}^2(\Omega)\cap\mathrm{H}^1_0(\Omega))'$. Since $\Omega$ as a Cartesian product of open intervals (or more generally of *convex* low-dimensional domains) is convex this is indeed the case.

Adhering to the basic idea in [3,4], we transform the variational problem first into an equivalent problem over the space $\ell_2(\nabla^d)$, where $\nabla$ is a countable index set. This allows us to employ *subspace-based tensor formats* for representing corresponding coefficient sequences as order-$d$ tensors. For the convergence of the tensors in $\ell_2(\nabla^d)$ to correspond to convergence of the respective functions in $\mathrm{L}_2(\Omega)$, we now need to perform an *asymmetric* preconditioning to arrive at an ideal convergent iteration for the problem on the infinite-dimensional space $\ell_2(\nabla^d)$. This central issue is addressed in Section 2. Section 3 is devoted to the precise formulation of the new algorithm and its convergence and complexity analysis. Finally, in Section 4, the theoretical findings are illustrated and quantified by numerical experiments.

Despite the loss of symmetry, the overall effect of the asymmetric preconditioning on the computational complexity turns out to be favorable, regarding both the theoretical complexity estimates (as summarized in Rem. 3.4) and the practical efficiency of the scheme – with the difference, of course, that errors are controlled in $\mathrm{L}_2(\Omega)$ and not in $\mathrm{H}^1(\Omega)$. The relevance of an $\mathrm{L}_2$-error control depends mainly on the particular application: in the context of implicit time-stepping schemes for high-dimensional parabolic problems, the $\mathrm{L}_2$-norm may generally be preferable; when solving high-dimensional eigenvalue problems using inverse iteration, however, controlling errors in $\mathrm{H}^1$ is crucial for ensuring the convergence of eigenvalues. Moreover, our approach extends to singularly perturbed problems of reaction-diffusion type $-\varepsilon\Delta u + u = f$ where, due to boundary layers for small $\varepsilon$, an $\mathrm{L}_2$-error bound is more meaningful.

We close this section with recalling some primarily technical preliminaries from [3, 4], where a more self-contained exposition can be found.

## 1.1. Prerequisites

*1.1.1. Tensor representations*

For simplicity of exposition, in what follows we focus as in [3] on problems of the form (1.1) with a constant diffusion matrix $M$ and $\Omega = (0,1)^d$ so that the operator

$$Au := -\sum_{i,j=1}^{d} m_{ij}\partial_i\partial_j u, \tag{1.4}$$

with constant coefficients $m_{ij}$ and symmetric positive definite $M = (m_{ij}) \in \mathbb{R}^{d\times d}$ satisfies (1.3). Furthermore, to avoid certain technicalities, we impose the slightly stronger assumption that $M$ is *diagonally dominant*.

In order to transform (1.1) into an equivalent problem over sequence spaces, we employ a tensor product *wavelet basis*

$$\{\Psi_\nu := \psi_{\nu_1} \otimes \ldots \otimes \psi_{\nu_d} \colon \nu = (\nu_1, \ldots, \nu_d) \in \nabla^d\},$$

where $\{\psi_\nu\}_{\nu\in\nabla}$ is an orthonormal basis of $\mathrm{L}_2(0,1)$ and $\{2^{-2|\nu|}\psi_\nu\}_{\nu\in\nabla}$ is a Riesz basis of $\mathrm{H}^2(0,1)\cap\mathrm{H}^1_0(0,1)$. Note that this requires, in particular, that the wavelets vanish on $\partial\Omega$, that is, the univariate factor wavelets satisfy $\psi_\nu(0) = \psi_\nu(1) = 0$, $\nu \in \nabla$. The corresponding wavelet representation of $A$ is then given by the infinite matrix

$$\mathbf{T} := \big(\langle \Psi_\mu, A\Psi_\nu\rangle\big)_{\mu,\nu\in\nabla^d}. \tag{1.5}$$

Since the homogeneous boundary conditions are built into the basis $\{\Psi_\nu\}$, finding the solution $u$ of (1.1) is equivalent to finding its wavelet coefficient sequence

$$\mathbf{u} := (u_\nu)_{\nu\in\nabla^d}, \quad u_\nu := \langle u, \Psi_\nu\rangle := \int_\Omega u\,\Psi_\nu\,\mathrm{d}x. \tag{1.6}$$

Defining $\mathbf{g} = (\langle f, \Psi_\nu\rangle)_{\nu\in\nabla^d}$, the sequence $\mathbf{u}$, in turn, is the solution of

$$\mathbf{T}\mathbf{u} = \mathbf{g}. \tag{1.7}$$

Thus our objective is to solve (1.7). Note that the operator $\mathbf{T}$ is unbounded as an operator from $\ell_2(\nabla^d)$ to itself, where as usual $\ell_2(\nabla^d)$ is the space of square summable sequences over the index set $\nabla^d$ endowed with the norm

$$\|\mathbf{v}\| := \|\mathbf{v}\|_{\ell_2(\nabla^d)} := \left(\sum_{\nu\in\ell_2(\nabla^d)} |v_\nu|^2\right)^{1/2}.$$

For the moment we postpone the discussion of the choice of subspace of $\ell_2(\nabla^d)$ for which (1.7) is supposed to hold and explain first some algebraic features of (1.7). Since $\nabla^d$ is a product set, we view any element $\mathbf{v} \in \ell_2(\nabla^d)$ as *a tensor of order d*. As an operator acting on such tensors, $\mathbf{T}$ has finite rank. More precisely, as has been pointed out in [3], $\mathbf{T}$ has the tensor representation

$$\mathbf{T} = \sum_{1\le n_1,\ldots,n_d\le R} c_{n_1,\ldots,n_d} \bigotimes_i \mathbf{T}^{(i)}_{n_i}, \tag{1.8}$$

with $R = 4$, and

$$\mathbf{T}^{(i)}_1 := \mathbf{T}_1 = \big(\langle\psi_\nu, \psi_\mu\rangle\big)_{\mu,\nu\in\nabla} = \mathrm{id}, \qquad \mathbf{T}^{(i)}_2 := \mathbf{T}_2 := \big(\langle\psi'_\nu, \psi'_\mu\rangle\big)_{\mu,\nu\in\nabla}, \tag{1.9}$$

$$\mathbf{T}^{(i)}_3 := \mathbf{T}_3 := \big(\langle\psi_\nu, \psi'_\mu\rangle\big)_{\mu,\nu\in\nabla}, \qquad \mathbf{T}^{(i)}_4 := \mathbf{T}_4 := \big(\langle\psi'_\nu, \psi_\mu\rangle\big)_{\mu,\nu\in\nabla}. \tag{1.10}$$

Here the nonzero entries $c_{n_1,\dots,n_d}$ of the sparse coefficient tensor $\mathbf{c}$ are given by $c_{2,1,\dots,1} = m_{11}$, $c_{1,2,1,\dots,1} = m_{22}$, $\dots$, $c_{3,4,1,\dots,1} = c_{4,3,1,\dots,1} = m_{12}$, $\dots$, $c_{3,1,4,1,\dots,1} = c_{4,1,3,1,\dots,1} = m_{13}$, and so forth (*cf.* [3], Sect. 2.2). Thus, for $A$ as in (1.4), in general we have that $R = 4$. Note further that, due to the homogeneous Dirichlet boundary conditions, integration by parts shows $\mathbf{T}_3 = -\mathbf{T}_4$, which gives

$$\mathbf{T}_3 \otimes \mathbf{T}_4 = \mathbf{T}_4 \otimes \mathbf{T}_3 = -\mathbf{T}_3 \otimes \mathbf{T}_3, \tag{1.11}$$

and thus a reduction to $R = 3$.

We shall now introduce some basic notions of tensor representations. For further details and references, we refer to [19]. The particular representation format for the operator $\mathbf{T}$ chosen in (1.8) corresponds to the so-called *Tucker format* for tensors of order $d$. Accordingly, as mentioned earlier, regarding $\mathbf{u}$ as a tensor of order $d$ on $\nabla^d = \bigtimes_{i=1}^d \nabla$, it can be represented in terms of the Tucker format

$$\mathbf{u} = \sum_{k_1=1}^{r_1} \dots \sum_{k_d=1}^{r_d} a_{k_1,\dots,k_d}\, \mathbf{U}^{(1)}_{k_1} \otimes \dots \otimes \mathbf{U}^{(d)}_{k_d}, \tag{1.12}$$

where $\mathbf{a} = (a_{k_1,\dots,k_d})_{1\le k_i\le r_i : i=1,\dots,d}$ is called the *core tensor* and each matrix $\mathbf{U}^{(i)} = \big(\mathbf{U}^{(i)}_{\nu_i,k_i}\big)_{\nu_i\in\nabla^{d_i},\,1\le k_i\le r_i}$ with orthonormal column vectors $\mathbf{U}^{(i)}_k \in \ell_2(\nabla^{d_i})$, $k = 1,\dots,r_i$, is called the $i$th *orthonormal mode frame* (here we admit $r_i = \infty, i = 1,\dots,d$). We refer to [3,4] for the precise definitions and notation, to which we will adhere in this paper as well. Of course, for an operator $\mathbf{T}$ on $\ell_2(\nabla^d)$ and an element $\mathbf{u} \in \ell_2(\nabla^d)$ that are both given as representations in the Tucker format, the image $\mathbf{Tu}$ can readily be expressed in the Tucker format [30] by a combination of the core tensors and the application of the $\mathbf{T}^{(i)}_k$ to the mode frames $\mathbf{U}^{(i)}_{k_i}$, see [4] for details.

Since the core tensor $\mathbf{a}$ in (1.12) still depends on $d$ indices, for large $d$ it will generally have far too many entries for a direct representation. For this reason, we focus in what follows on the *hierarchical Tucker format* [21], which is obtained by further decomposing $\mathbf{a}$ into successive compositions of third-order tensors as

$$\mathbf{a} = \Big(\Sigma_{\mathcal{D}_d}\big(\{\mathbf{B}^{(\alpha,k)}\}\big)\Big)_{(k_\beta)_{\beta\in\mathcal{L}(\mathcal{D}_d)}} := \sum_{(k_\gamma)_{\gamma\in\mathcal{I}(\mathcal{D}_d)}} \prod_{\delta\in\mathcal{N}(\mathcal{D}_d)} B^{(\delta,k_\delta)}_{(k_{\mathrm{c}_1(\delta)},k_{\mathrm{c}_2(\delta)})}.$$

This is based on a fixed *binary dimension tree* $\mathcal{D}_d$ obtained by successive bisections of the set of coordinate indices $0_d := \{1,\dots,d\}$, which forms the root node. Moreover, singletons $\{i\} \in \mathcal{D}_d$ are referred to as *leaves*, and elements of $\mathcal{D}_d \setminus \big\{0_d, \{1\},\dots,\{d\}\big\}$ as *interior nodes*. The set of leaves is denoted by $\mathcal{L}(\mathcal{D}_d)$, where we additionally set $\mathcal{N}(\mathcal{D}_d) := \mathcal{D}_d \setminus \mathcal{L}(\mathcal{D}_d)$. The functions

$$\mathrm{c}_i : \mathcal{D}_d \setminus \mathcal{L}(\mathcal{D}_d) \to \mathcal{D}_d \setminus \{0_d\}, \qquad i = 1,2,$$

produce the "left" and "right" children $\mathrm{c}_i(\alpha) \subset \alpha$ of a non-leaf node $\alpha \in \mathcal{N}(\mathcal{D}_d)$.

With each node $\alpha \in \mathcal{D}_d$ we associate the *matricization* $T^{(\alpha)}_{\mathbf{u}}$ of $\mathbf{u}$, obtained by rearranging the entries of the tensor into an infinite matrix representation of a Hilbert–Schmidt operator using the indices in $\nabla^\alpha$ as row indices. The dimensions of the ranges of these operators yield the *hierarchical ranks* $\mathrm{rank}_\alpha(\mathbf{u}) := \dim \mathrm{range}\, T^{(\alpha)}_{\mathbf{u}}$ for $\alpha \in \mathcal{D}_d$. Except for $\alpha = 0_d$, where we always have $\mathrm{rank}_{0_d}(\mathbf{u}) \le 1$, these are collected in the *hierarchical rank vector* $\mathrm{rank}(\mathbf{u}) = \mathrm{rank}_{\mathcal{D}_d}(\mathbf{u}) := (\mathrm{rank}_\alpha(\mathbf{u}))_{\alpha\in\mathcal{D}_d\setminus\{0_d\}}$ and give rise to the hierarchical tensor classes

$$\mathcal{H}(\mathsf{r}) := \big\{\mathbf{u} \in \ell_2(\nabla^d) \colon \mathrm{rank}_\alpha(\mathbf{u}) \le r_\alpha \text{ for all } \alpha \in \mathcal{D}_d \setminus \{0_d\}\big\}.$$

For singletons $\{i\} \in \mathcal{D}_d$, we briefly write $\mathrm{rank}_i(\mathbf{u}) := \mathrm{rank}_{\{i\}}(\mathbf{u})$. We denote by $\mathcal{R} \subset (\mathbb{N}_0 \cup \{\infty\})^{\mathcal{D}_d\setminus\{0_d\}}$ the set of hierarchical rank vectors $\mathsf{r}$ for which there exists $\mathbf{u}$ such that $\mathrm{rank}(\mathbf{u}) = \mathsf{r}$.

Again, there is an analogous hierarchical format for operators, *i.e.*, the core tensor $\mathbf{c}$ in (1.8) is further decomposed as a product of tensors of order three, and the format is consistent when applying an operator to

a tensor, see [4]. The hierarchical ranks in the representation of $\mathbf{c}$ will be denoted by $R_\alpha$, $\alpha \in \mathcal{D}_d$. In what follows we are mainly interested in two scenarios, namely that $M = \operatorname{diag}(m_{ii})_{i=1}^d$ for some $m_{ii} > 0$, or that $M$ is tridiagonal with constant diagonal vectors. In the former case, we have $R = 2$ as well as $R_\alpha \le 2$. For tridiagonal $M$, in general one obtains $R = 4$ and $R_\alpha \le 5$, but in the present case, due to (1.11), this reduces to $R = 3$ and $R_\alpha \le 4$. We refer to ([3], Ex. 3.2) for more details.

*1.1.2. Recompression and coarsening*

The basic strategy suggested in [3, 4], which we follow here as well, is to solve (1.7) iteratively. At a first glance this looks promising since the application of the finite-rank operator $\mathbf{T}$ to a finite-rank iterate produces (at least for a suitably truncated finite-rank right hand side) a new iterate of finite rank. As mentioned earlier, at least two principal obstructions arise. First, the action of the operator as well as the summation of finite rank tensors increase the tensor ranks in each step, so that a straightforward iteration would give rise to exponentially increasing ranks. Second, increasing the ranks of tensor expansions has to go hand in hand with growing the supports of increasingly more accurately resolved mode frames. In this section we briefly recall from [3,4] how to deal with these issues. The key point is to devise suitable tensor recompression and coarsening schemes that automatically find near-best approximations from the classes $\mathcal{H}(\mathsf{r})$ whose mode frames have near-minimal supports. Again we refer to [4] for a detailed derivation and recall here the main results for later use. The *hierarchical singular value decomposition* ($\mathcal{H}$SVD) (*cf.* [16]) allows one to identify for given tensor $\mathbf{v}$ a system of mode frames, denoted by $\mathbb{U}(\mathbf{v})$, whose rank truncation yields near-optimal approximations. We denote by $\mathrm{P}_{\mathbb{U}(\mathbf{v}),\mathsf{r}}\,\mathbf{v}$ the result of truncating a $\mathcal{H}$SVD of $\mathbf{v}$ to ranks $\mathsf{r}$. Using computable upper bounds $\lambda_{\mathsf{r}}(\mathbf{v})$ for $\|\mathbf{v} - \mathrm{P}_{\mathbb{U}(\mathbf{v}),\mathsf{r}}\,\mathbf{v}\|$, one can determine ranks $\mathrm{r}(\mathbf{u},\eta) \in \arg\min\{|\mathsf{r}|_\infty : \mathsf{r} \in \mathcal{R},\ \lambda_{\mathsf{r}}(\mathbf{u}) \le \eta\}$ that ensure the validity of a given accuracy tolerance $\eta > 0$, which we use to define the recompression operator $\hat{\mathrm{P}}_\eta\,\mathbf{v} := \mathrm{P}_{\mathbb{U}(\mathbf{v}),\mathrm{r}(\mathbf{v},\eta)}\,\mathbf{v}$.

The definition of a *coarsening* operator producing near-minimal supports of mode frames in a sense to be made precise later, is a little more involved and based on the notion of *tensor contractions* which, for $i = 1, \dots, d$, are given by

$$\pi^{(i)}(\mathbf{v}) = \big(\pi^{(i)}_{\nu_i}(\mathbf{v})\big)_{\nu_i\in\nabla} := \left(\left(\sum_{\nu_1,\dots,\nu_{i-1},\nu_{i+1},\dots,\nu_d} |v_{\nu_1,\dots,\nu_{i-1},\nu_i,\nu_{i+1},\dots,\nu_d}|^2\right)^{\frac12}\right)_{\nu_i\in\nabla} \in \ell_2(\nabla).$$

A naive evaluation of these quantities requires a $(d-1)$-dimensional summation, which would be inacceptable. However, the identity

$$\pi^{(i)}_\nu(\mathbf{v}) = \left(\sum_k \big|\mathbf{U}^{(i)}_{\nu,k}\big|^2 \big|\sigma^{(i)}_k\big|^2\right)^{\frac12},$$

where $\sigma^{(i)}_k$ are the mode-$i$ singular values and $\{\mathbf{U}^{(i)}_k\}$ the corresponding mode frames from $\mathbb{U}(\mathbf{v})$, facilitates an evaluation at a cost proportional to $\operatorname{rank}_i(\mathbf{v})$ for each $\nu \in \ell_2(\nabla)$, see [4]. The quantities

$$\operatorname{supp}_i(\mathbf{v}) := \operatorname{supp}\big(\pi^{(i)}(\mathbf{v})\big)$$

allow one to quantify the the actual number of nonzero entries of mode frames, and we have $\operatorname{supp}\mathbf{v} \subseteq \times_{i=1}^d \operatorname{supp}_i(\mathbf{v})$. With the aid of a total ordering of the entries of all $\pi^{(i)}(\mathbf{v})$, $i = 1, \dots, d$, one can find for a given $\mathbf{v}$ a product set $\Lambda(\mathbf{v};N)$, with sum of coordinatewise cardinalities at most $N$, such that the restriction $\mathrm{R}_{\Lambda(\mathbf{v};N)}$ of $\mathbf{v}$ to $\Lambda(\mathbf{v};N)$ (meaning that the entries $v_\nu$ are set to zero for $\nu \notin \Lambda(\mathbf{v};N)$) satisfies

$$\|\mathbf{v} - \mathrm{R}_{\Lambda(\mathbf{v};N)}\,\mathbf{v}\| \le \mu_N(\mathbf{v}) \le \sqrt{d}\,\inf\{\|\mathbf{v} - \mathrm{R}_{\hat\Lambda}\,\mathbf{v}\| : \hat\Lambda = \hat\Lambda_1 \times \ldots \times \hat\Lambda_d,\ \textstyle\sum_i \#(\hat\Lambda_i) \le N\},$$

where the error estimate $\mu_N(\mathbf{v})$ can be computed directly from the sequences $\pi^{(i)}(\mathbf{v})$. Setting $N(\mathbf{v},\eta) := \min\{N : \mu_N(\mathbf{v}) \le \eta\}$, we define the *thresholding* procedure

$$\hat{\mathrm{C}}_\eta(\mathbf{v}) := \mathrm{R}_{\Lambda(\mathbf{v};N(\mathbf{v};\eta))}\,\mathbf{v}. \tag{1.13}$$

To assess the performance of the recompression and coarsening operators $\hat{\mathrm{P}}_\eta, \hat{\mathrm{C}}_\eta$, as in [4] we define

$$\sigma_{r,\mathcal{H}}(\mathbf{v}) := \inf\big\{\|\mathbf{v}-\mathbf{w}\| \,:\, \mathbf{w}\in\mathcal{H}(\mathsf{r}) \text{ with } \mathsf{r}\in\mathcal{R},\ |\mathsf{r}|_\infty \le r\big\},$$

and, for a given *growth sequence* $\gamma = \big(\gamma(n)\big)_{n\in\mathbb{N}_0}$ with $\gamma(0)=1$ and $\gamma(n)\to\infty$ as $n\to\infty$, we consider

$$\mathcal{A}(\gamma) = \mathcal{A}_{\mathcal{H}}(\gamma) := \Big\{\mathbf{v}\in\ell_2(\nabla^d) : \sup_{r\in\mathbb{N}_0}\gamma(r)\,\sigma_{r,\mathcal{H}}(\mathbf{v}) =: |\mathbf{v}|_{\mathcal{A}_{\mathcal{H}}(\gamma)} < \infty\Big\},$$

where we set $\|\mathbf{v}\|_{\mathcal{A}_{\mathcal{H}}(\gamma)} := \|\mathbf{v}\| + |\mathbf{v}|_{\mathcal{A}_{\mathcal{H}}(\gamma)}$. We always require that $\rho_\gamma := \sup_{n\in\mathbb{N}}\gamma(n)/\gamma(n-1) < \infty$, which covers at most exponential growth. Thus, hierarchical ranks of size at most $\gamma^{-1}(|\mathbf{v}|_{\mathcal{A}_{\mathcal{H}}(\gamma)}/\eta)$ suffice to approximate $\mathbf{v}\in\mathcal{A}(\gamma)$ within accuracy $\eta$.

Similarly, defining the error of best $N$-term approximation

$$\sigma_N(\mathbf{v}) := \inf_{\substack{\Lambda\subset\nabla^{\hat d}\\ \#\Lambda\le N}} \|\mathbf{v}-\mathrm{R}_\Lambda\,\mathbf{v}\|,$$

we consider for $s>0$ the classical *approximation classes* $\mathcal{A}^s = \mathcal{A}^s(\nabla^{\hat d})$, $\hat d\in\mathbb{N}$, comprised of all $\mathbf{v}\in\ell_2(\nabla^{\hat d})$ for which the quasi-norm

$$\|\mathbf{v}\|_{\mathcal{A}^s(\nabla^{\hat d})} := \sup_{N\in\mathbb{N}_0}(N+1)^s\sigma_N(\mathbf{v})$$

is finite. Hence, using this concept for $\hat d = 1$, when the mode frames belong to $\mathcal{A}^s(\nabla)$, they can be approximated within accuracy $\eta$ by finitely supported vectors of size $\mathcal{O}(\eta^{-1/s})$.

The relevant facts describing the performance of $\hat{\mathrm{P}}_\eta$ and $\hat{\mathrm{C}}_\eta$ can be summarized as follows [4].

**Theorem 1.1.** *Let $\mathbf{u},\mathbf{v}\in\ell_2(\nabla^d)$ with $\mathbf{u}\in\mathcal{A}_{\mathcal{H}}(\gamma)$, $\pi^{(i)}(\mathbf{u})\in\mathcal{A}^s$ for $i=1,\dots,d$, and $\|\mathbf{u}-\mathbf{v}\|\le\eta$. Let $\kappa_{\mathrm{P}} = \sqrt{2d-3}$ and $\kappa_{\mathrm{C}} = \sqrt{d}$. Then, for any fixed $\alpha>0$,*

$$\mathbf{w}_\eta := \hat{\mathrm{C}}_{\kappa_{\mathrm{C}}(\kappa_{\mathrm{P}}+1)(1+\alpha)\eta}\big(\hat{\mathrm{P}}_{\kappa_{\mathrm{P}}(1+\alpha)\eta}(\mathbf{v})\big)$$

*satisfies*

$$\|\mathbf{u}-\mathbf{w}_\eta\| \le C(\alpha,\kappa_{\mathrm{P}},\kappa_{\mathrm{C}})\,\eta, \tag{1.14}$$

*where $C(\alpha,\kappa_{\mathrm{P}},\kappa_{\mathrm{C}}) := \big(1+\kappa_{\mathrm{P}}(1+\alpha)+\kappa_{\mathrm{C}}(\kappa_{\mathrm{P}}+1)(1+\alpha)\big)$, as well as*

$$|\mathrm{rank}(\mathbf{w}_\eta)|_\infty \le \gamma^{-1}\big(\rho_\gamma\|\mathbf{u}\|_{\mathcal{A}_{\mathcal{H}}(\gamma)}/(\alpha\eta)\big), \qquad \|\mathbf{w}_\eta\|_{\mathcal{A}_{\mathcal{H}}(\gamma)} \le C_1\|\mathbf{u}\|_{\mathcal{A}_{\mathcal{H}}(\gamma)}, \tag{1.15}$$

*with $C_1 = (\alpha^{-1}(1+\kappa_{\mathrm{P}}(1+\alpha))+1)$ and*

$$\begin{aligned}
&\sum_{i=1}^d \#\,\mathrm{supp}_i(\mathbf{w}_\eta) \le 2\eta^{-\frac{1}{s}}d\,\alpha^{-\frac{1}{s}}\left(\sum_{i=1}^d\|\pi^{(i)}(\mathbf{u})\|_{\mathcal{A}^s}\right)^{\frac{1}{s}},\\
&\sum_{i=1}^d\|\pi^{(i)}(\mathbf{w}_\eta)\|_{\mathcal{A}^s} \le C_2\sum_{i=1}^d\|\pi^{(i)}(\mathbf{u})\|_{\mathcal{A}^s},
\end{aligned} \tag{1.16}$$

*with $C_2 = 2^s(1+3^s)+2^{4s}\alpha^{-1}\big(1+\kappa_{\mathrm{P}}(1+\alpha)+\kappa_{\mathrm{C}}(\kappa_{\mathrm{P}}+1)(1+\alpha)\big)d^{\max\{1,s\}}$.*

**Remark 1.2.** Both $\hat{\mathrm{P}}_\eta$ and $\hat{\mathrm{C}}_\eta$ require a hierarchical singular value decomposition of their inputs. For a finitely supported $\mathbf{v}$ given in hierarchical format, the number of operations required for obtaining such a decomposition is bounded, up to a fixed multiplicative constant, by $d|\mathrm{rank}(\mathbf{v})|_\infty^4 + |\mathrm{rank}(\mathbf{v})|_\infty^2\sum_{i=1}^d\#\,\mathrm{supp}_i\,\mathbf{v}$, see also [16].

## 2. Asymmetric preconditioning

### 2.1. Transformation to well-conditioned systems

Following [3, 4], to iteratively solve (1.7) and hence (1.1), we first need to precondition the operator $\mathbf{T}$ to obtain a well-conditioned operator equation on $\ell_2(\nabla^d)$. A natural way of doing this is to exploit the mapping properties (1.3) in combination with the fact that a suitable diagonal scaling of the $L_2(\Omega)$-wavelet basis gives rise to a *Riesz basis* for $H^1_0(\Omega)$. To describe this we choose for $i = 1, \dots, d$ the scaling weights $\hat{\omega}_{i,\nu_i}$, $\nu_i \in \nabla$, such that

$$\hat{\omega}_{i,\nu_i} \sim 2^{|\nu_i|} \tag{2.1}$$

where the constants are uniform in $\nu_i \in \nabla$, and set

$$\omega_\nu := \omega_{\nu_1,\dots,\nu_d} = \left(\sum_{i=1}^d (\hat{\omega}_{i,\nu_i})^2\right)^{1/2}, \quad \nu \in \nabla^d. \tag{2.2}$$

With this sequence, we define the diagonal scaling operator

$$\mathbf{S} = \left(\omega_\nu \delta_{\nu,\mu}\right)_{\nu,\mu\in\nabla^d}. \tag{2.3}$$

In addition, for later reference, we define for $\tau \in \mathbb{R}$ and $i = 1, \dots, d$ on the one hand the coordinatewise scaling operators $\mathbf{S}_i^\tau : \mathbb{R}^{\nabla^d} \to \mathbb{R}^{\nabla^d}$ by

$$\mathbf{S}_i^\tau \mathbf{v} := \left(\hat{\omega}_{i,\nu_i}^\tau v_\nu\right)_{\nu\in\nabla^d} \quad \text{and} \quad \mathbf{S}_i := \mathbf{S}_i^1, \tag{2.4}$$

and on the other hand, the corresponding *low-dimensional* scaling operators $\hat{\mathbf{S}}_i^\tau : \mathbb{R}^\nabla \to \mathbb{R}^\nabla$ by

$$\hat{\mathbf{S}}_i^\tau \hat{\mathbf{v}} := \left(\hat{\omega}_{i,\nu_i}^\tau \hat{v}_{\nu_i}\right)_{\nu_i\in\nabla} \quad \text{and} \quad \hat{\mathbf{S}}_i := \hat{\mathbf{S}}_i^1. \tag{2.5}$$

It is well-known that under the above assumptions on the basis $\{\Psi_\nu\}$, the rescaled mapping $\mathbf{S}^{-1}\mathbf{T}\mathbf{S}^{-1}$ is an isomorphism from $\ell_2(\nabla^d)$ onto itself, which is related to the fact that $u \in H^1_0(\Omega)$ if and only if $\mathbf{S}\mathbf{u} \in \ell_2(\nabla^d)$, where $\mathbf{u}$ is the wavelet coefficient tensor with respect to the $L_2(\Omega)$-basis. Note that this implies, in particular, that for each $\nu \in \nabla^d$ the quantity $(\mathbf{T}\mathbf{u})_\nu$ is well-defined when the corresponding function $u$ belongs to $H^1_0(\Omega)$. These facts have been exploited in [3] by replacing (1.7) by the (symmetrically) preconditioned system $\mathbf{S}^{-1}\mathbf{T}\mathbf{S}^{-1}\mathbf{u}^{(1)} = \mathbf{S}^{-1}\mathbf{g}$, *i.e.*, one actually solves for the $H^1(\Omega)$-scaled coefficient array $\mathbf{u}^{(1)} = \mathbf{S}\mathbf{u}$.

In this paper we follow a different direction, seeking directly the $L_2(\Omega)$-wavelet coefficients $\langle u, \Psi_\nu\rangle$ of the solution $u$ to (1.1), and thus of (1.7). Here we exploit that $(\mathbf{T}\mathbf{u})_\nu$ is, for every $\nu \in \nabla^d$, still well-defined for arbitrary $\mathbf{u} \in \ell_2(\nabla^d)$ provided that the wavelet basis functions are sufficiently regular. Our approach is based on the following facts.

**Theorem 2.1.** *Assume that the univariate wavelet basis $\{\psi_\nu\}$ is $L_2(0,1)$-orthonormal and that $\{2^{-2|\nu|}\psi_\nu\}$ is a Riesz basis of $H^2(0,1)\cap H^1_0(0,1)$. Then, for $\mathbf{S}, \mathbf{T}$ defined by (2.3), (1.8), respectively, the infinite matrix $\mathbf{S}^{-2}\mathbf{T}$ is an isomorpishm from $\ell_2(\nabla^d)$ onto itself,* i.e., *there exist constants $0 < c \le C < \infty$ such that*

$$c\|\mathbf{v}\| \le \|\mathbf{S}^{-2}\mathbf{T}\mathbf{v}\| \le C\|\mathbf{v}\|, \quad \mathbf{v} \in \ell_2(\nabla^d). \tag{2.6}$$

*Moreover, when $M$ is diagonal and $\hat{\omega}_{i,\nu_i} \sim \sqrt{m_{ii}}\, 2^{|\nu_i|}$ with constants that are uniform in $d$, $i = 1, \dots, d$ and $\nu_i$, then the constants $c, C$ are independent of the spatial dimension $d$.*

**Remark 2.2.** The property that the univariate rescaled wavelet basis is a Riesz basis for $H^2(0,1)\cap H^1_0(0,1)$, required in Theorem 2.1, is satisfied, in particular, under the following conditions: the univariate wavelet (or multiwavelet) basis functions are $L_2$-orthonormal, piecewise polynomial, belong to $C^1(0,1)$, vanish at the endpoints of the interval, and the scaling functions have the *polynomial reproduction property*. To describe this

latter property consider any closed subinterval of $[0,1]$ on which the scaling functions of that level are polynomial. Then, for each level and any such subinterval contained in the interior, all polynomials of degree two are reproduced. On those such subintervals containing an endpoint of $[0,1]$, only those polynomials are reproduced that vanish at that endpoint. Note that a piecewise polynomial in $C^1(0,1)$ belongs to $H^{2+\tau}(0,1)$ for any $\tau < \frac{1}{2}$. With the above properties the validity of suitable inverse and direct estimates can be verified which, combined with orthogonality, imply the required Riesz basis property, see [10].

*Proof.* Note that $V := H_0^1(\Omega) \cap H^2(\Omega) = \bigcap_{i=1}^d L_2(0,1) \otimes \ldots \otimes (H_0^1(0,1) \cap H^2(0,1)) \otimes \ldots \otimes L_2(0,1)$. Thus the rescaled wavelets $\omega_\nu^{-2}\Psi_\nu$, $\nu \in \nabla^d$, form a Riesz basis for $V$, and they are therefore the dual of a Riesz-basis for $V' = (H_0^1(\Omega) \cap H^2(\Omega))'$, *i.e.*, for $w \in V'$ one has $\|w\|_{V'} \sim \|(\langle w, \omega_\nu^{-2}\Psi_\nu\rangle)_{\nu\in\nabla^d}\|_{\ell_2(\nabla^d)}$. Due to the convexity of $\Omega$ and the fact that $M$ is constant, the operator $A$ maps $V$ one-to-one and onto $L_2(\Omega)$ and hence, by duality, from $L_2(\Omega)$ onto $V'$. Since $(\langle Au, \omega_\nu^{-2}\Psi_\nu\rangle)_{\nu\in\nabla^d} = \mathbf{S}^{-2}\mathbf{T}\mathbf{u}$ and $\|Au\|_{V'} \sim \|u\|_{L_2(\Omega)} \sim \|\mathbf{u}\|_{\ell_2(\nabla^d)}$, the norm equivalence (2.6) follows.

To prove the rest of the assertion for diagonal $M$, by the choice of $\hat\omega_{i,\nu_i}$ (*cf.* [3,12]), it suffices to confine the discussion to the Laplacian on $H_0^1(\Omega)$, where

$$\mathbf{T} = \sum_{i=1}^d \mathbf{T}_2^{(i)} := \sum_{i=1}^d \mathrm{id}_1 \otimes \ldots \otimes \mathrm{id}_{i-1} \otimes \mathbf{T}_2 \otimes \mathrm{id}_{i+1} \otimes \ldots \otimes \mathrm{id}_d$$

with $\mathbf{T}_2$ as in (1.9). In order to estimate the constants $c, C$ in (2.6) in this case, we hence need to find bounds for the extreme singular values of $\mathbf{S}^{-2}\mathbf{T}$, or equivalently, the eigenvalues of $\mathbf{S}^{-2}\mathbf{T}\mathbf{T}^*\mathbf{S}^{-2} = \mathbf{S}^{-2}\mathbf{T}^2\mathbf{S}^{-2}$. To this end, recall that

$$\mathbf{S}^2 = \sum_{i=1}^d \mathbf{S}_i^2 = \sum_{i=1}^d \mathrm{id}_1 \otimes \ldots \otimes \mathrm{id}_{i-1} \otimes \hat{\mathbf{S}}_i^2 \otimes \mathrm{id}_{i+1} \otimes \ldots \otimes \mathrm{id}_d.$$

The desired statement follows if we can show that, for any *compactly supported* $\mathbf{v}$,

$$c^2\langle \mathbf{S}^4\mathbf{v}, \mathbf{v}\rangle \le \langle \mathbf{T}^2\mathbf{v}, \mathbf{v}\rangle \le C^2\langle \mathbf{S}^4\mathbf{v}, \mathbf{v}\rangle \tag{2.7}$$

with suitable $c, C$, since then the singular values of $\mathbf{S}^{-2}\mathbf{T}$ are contained in $[c, C]$.

We now estimate the summands in the expansions

$$\mathbf{S}^4 = \sum_{i,j=1}^d \mathbf{S}_i^2\mathbf{S}_j^2, \quad \mathbf{T}^2 = \sum_{i,j=1}^d \mathbf{T}_2^{(i)}\mathbf{T}_2^{(j)}$$

separately and then add the different contributions to obtain (2.7) with $c, C$ independent of $d$. If $i \ne j$, we have $\tilde c, \tilde C$ such that

$$\tilde c^2\, \hat{\mathbf{S}}_i^2 \otimes \hat{\mathbf{S}}_j^2 \le \mathbf{T}_2 \otimes \mathbf{T}_2 \le \tilde C^2\, \hat{\mathbf{S}}_i^2 \otimes \hat{\mathbf{S}}_j^2$$

in the sense, analogously to (2.7), of inner products with compactly supported sequences on $\nabla^2$; here we need only that $\{\psi_\nu\}$ is an orthonormal basis of $L_2(0,1)$ and $\{2^{-|\nu|}\psi_\nu\}$ is a Riesz basis of $H_0^1(0,1)$.

The case $i = j$ is, however, more involved: in general, we do *not* have $\tilde c^2\, \hat{\mathbf{S}}_i^4 \le \mathbf{T}_2^2 \le \tilde C^2\, \hat{\mathbf{S}}_i^4$ with the same $\tilde c, \tilde C$. Now we use in addition that $\{2^{-2|\nu|}\psi_\nu\}$ is a Riesz basis of $V_1 := H^2(0,1) \cap H_0^1(0,1)$. Using also $L_2$-orthonormality, we obtain

$$\mathbf{T}_{2,\mu\nu}^2 = \sum_\lambda \langle \psi_\mu', \psi_\lambda'\rangle\langle \psi_\lambda', \psi_\nu'\rangle = \sum_\lambda \langle \psi_\mu'', \psi_\lambda\rangle\langle \psi_\lambda, \psi_\nu''\rangle = \langle \psi_\mu'', \psi_\nu''\rangle.$$

We now verify that $\|u''\|_{L_2(0,1)}$ is a norm on $V_1$ by comparison with the standard norm $\|u\|_{V_1}^2 := \|u\|_{L_2(0,1)}^2 + \|u'\|_{L_2(0,1)}^2 + \|u''\|_{L_2(0,1)}^2$. By the Poincaré inequality, $\|u''\|_{L_2(0,1)} \gtrsim \|u' - \int_0^1 u'\,dx\|_{L_2(0,1)} = \|u'\|_{L_2(0,1)}$, where we have used that $\int_0^1 u'\,dx = 0$ as a consequence of $u \in H_0^1(0,1)$. By the Poincaré–Friedrichs inequality,

$\|u'\|_{L_2(0,1)} \gtrsim \|u\|_{L_2(0,1)}$. Hence $\|u''\|_{L_2(0,1)} \sim \|u\|_{V_1}$. By the Riesz basis property for $V_1$, we thus have $\hat{c}, \hat{C}$ such that

$$\hat{c}^2\, \hat{\mathbf{S}}_i^4 \le \mathbf{T}_2^2 \le \hat{C}^2\, \hat{\mathbf{S}}_i^4.$$

We thus obtain (2.7) with $c = \min\{\tilde{c}, \hat{c}\}$, $C = \max\{\tilde{c}, \hat{C}\}$, which are in particular independent of $d$. □

**Remark 2.3.** Clearly, unlike the symmetrically preconditioned version $\mathbf{S}^{-1}\mathbf{T}\mathbf{S}^{-1}$ considered in [3], $\mathbf{S}^{-2}\mathbf{T}$ is in general nonsymmetric. It is generally also nonnormal, since normality would require $\|\mathbf{S}^{-2}\mathbf{T}\mathbf{v}\| = \|\mathbf{T}\mathbf{S}^{-2}\mathbf{v}\|$ for any $\mathbf{v} \in \ell_2(\nabla^d)$ and hence, in particular,

$$\sum_\mu \big(\omega_\mu^{-2} T_{\mu\nu}\big)^2 = \omega_\nu^{-4} \sum_\mu \big(T_{\mu\nu}\big)^2, \quad \nu \in \nabla^d,$$

which holds only for very specific choices of $\{\Psi_\nu\}$ (*e.g.*, for a basis of eigenfunctions of $A$).

**Remark 2.4.** As a consequence of Theorem 2.1, for suitable $\omega > 0$ the Jacobi-type iteration

$$\mathbf{u}_{j+1} = \mathbf{u}_j - \omega \mathbf{S}^{-2}(\mathbf{T}\mathbf{u}_j - \mathbf{g}) \tag{2.8}$$

converges in $\ell_2(\nabla^d)$, so that the corresponding wavelet expansions $u_j$ converge in $L_2(\Omega)$. The desired error control in $L_2$, with all approximations performed with respect to an equivalent norm, therefore *requires* the asymmetric preconditioning, whereas the symmetrically preconditioned iteration considered in [3] leads to control of all arising errors in $H^1$-norm.

Our envisaged numerical scheme may be viewed as a *perturbed* version of the iteration (2.8). The perturbations result from approximating all quantities by finitely supported sequences and from additional low-rank approximations in hierarchical tensor format. While the asymmetric preconditioning by $\mathbf{S}^{-2}$ causes the loss of symmetry it has the following advantage: the application of the finite-rank operator $\mathbf{T}$ to a finite-rank iterate $\mathbf{u}_j$ increases the output rank by at most a factor $\max_{\alpha\in\mathcal{D}_d} R_\alpha$, which is determined by the diffusion matrix $M$ and can therefore implicitly depend on $d$. Concerning cases of interest where this factor is also independent of $d$, we refer to the discussion at the end of Section 1.1.1. The scaling operator $\mathbf{S}^{-2}$, however, has *infinite rank* so that the construction of a finite-rank approximation to the scaled residual $\mathbf{S}^{-2}(\mathbf{T}\mathbf{u}_j - \mathbf{g})$ must involve a substantial rank reduction. For finding a good compromise between accuracy and rank size, Theorem 1.1 is pivotal. Note that in the symmetric case $\mathbf{S}^{-1}\mathbf{T}\mathbf{S}^{-1}$, the rank-inflating scaling operation has to be done twice, with corresponding consequences concerning computational complexity, see Remark 3.4 below. The effect of a one-sided scaling will later be quantified, in addition to an analytical assessment, by our numerical experiments.

Our strategy for producing an approximate finite-rank residual is similar in spirit to the approach in [3] for the symmetric case, namely to approximate the scaling operator $\mathbf{S}^{-2}$ by a finite-rank operator. The foundation of this approximation is given in the next section.

## 2.2. Low-rank preconditioner

Rather than approximately applying $\mathbf{S}^{-1}$ twice we find a direct finite-rank approximation for $\mathbf{S}^{-2}$ with the aid of the following *relative error estimate for exponential sum approximation.*

**Theorem 2.5.** *Let* $\delta \in (0,1)$ *and*

$$0 < h \le \sup_{b\in(0,\pi/2)} \frac{2\pi b}{4 + |\ln\delta| + |\ln\cos b|}, \quad \alpha(x) := \ln(1+\mathrm{e}^x), \quad w(x) := (1+\mathrm{e}^{-x})^{-1}. \tag{2.9}$$

*Let* $n^+ := \lceil h^{-1}|\ln(\frac{1}{2}\delta)|\rceil$ *and*

$$\varphi_{h,n}(t) := \sum_{k=-n}^{n^+} h\, w(kh)\, \mathrm{e}^{-\alpha(kh)\, t}, \quad \varphi_{h,\infty}(t) := \lim_{n\to\infty} \varphi_{h,n}(t). \tag{2.10}$$

*Then*

$$\left|t^{-1} - \varphi_{h,\infty}(t)\right| \le \delta\, t^{-1} \quad \textit{for all } t \in [1,\infty), \tag{2.11}$$

*and furthermore, for any* $\varepsilon > 0$ *and* $n \ge \lceil h^{-1}|\ln \varepsilon|\rceil$, *we have*

$$\left|\varphi_{h,n}(t) - \varphi_{h,\infty}(t)\right| \le \varepsilon \quad \textit{for all } t \in [1,\infty). \tag{2.12}$$

*Consequently, for* $\eta > 0$, $T > 1$, *and* $n \ge \lceil h^{-1}(|\ln \eta| + \ln T)\rceil$, *we have*

$$\left|\varphi_{h,n}(t) - \varphi_{h,\infty}(t)\right| \le \eta\, t^{-1} \quad \textit{for all } t \in [1,T]. \tag{2.13}$$

Note that the supremum in (2.9) is attained for any $\delta > 0$.

*Proof.* Our starting point is the integral representation (*cf.* [20])

$$\frac{1}{r} = \int_0^\infty \mathrm{e}^{-rt}\,\mathrm{d}t = \int_{-\infty}^\infty \mathrm{e}^{-r\ln(1+\mathrm{e}^x)}\frac{\mathrm{d}x}{1+\mathrm{e}^{-x}}. \tag{2.14}$$

The integrand admits an analytic extension in the strip $\{x+iy\colon x\in\mathbb{R}, |y|<\pi/2\}$. Our aim is to apply ([29], Thm. 3.2.1), which gives

$$\left|\frac{1}{t} - \sum_{k\in\mathbb{Z}} h\,\omega(kh)\mathrm{e}^{-\alpha(kh)\,t}\right| \le N_b \frac{\mathrm{e}^{-\pi b/h}}{2\sinh(\pi b/h)},$$

where

$$N_b := \int_{\mathbb{R}} \left|\frac{\mathrm{e}^{-t\ln(1+\mathrm{e}^{x+ib})}}{1+\mathrm{e}^{-(x+ib)}}\right| \mathrm{d}x + \int_{\mathbb{R}} \left|\frac{\mathrm{e}^{-t\ln(1+\mathrm{e}^{x-ib})}}{1+\mathrm{e}^{-(x-ib)}}\right| \mathrm{d}x, \quad b\in(0,\pi/2).$$

We thus need a suitable estimate for $N_b$. Note that $|1+\mathrm{e}^{x\pm ib}|^2 \ge 1 + \mathrm{e}^{2x} \ge \frac{1}{2}(1+\mathrm{e}^x)^2$ and $|\ln(1+\mathrm{e}^{x\pm ib})| = \ln(1+\mathrm{e}^x\cos b)$. Furthermore, for $x\ge 0$ we obtain $1+\mathrm{e}^x\cos b \ge \mathrm{e}^{x\cos b}$ from comparing the respective series expansions, hence $\ln(1+\mathrm{e}^x\cos b)\ge x\cos b$ for $x\ge 0$. For $x\le 0$, we observe that $\ln(1+y)\ge \frac{1}{2}y$ for any $y\in[0,1]$, and hence $\ln(1+\mathrm{e}^x\cos b)\ge \frac{1}{2}x\cos b$ for $x\le 0$.

For such $b$, we now obtain

$$\int_{\mathbb{R}^+} \left|\frac{\mathrm{e}^{-t\ln(1+\mathrm{e}^{x\pm ib})}}{1+\mathrm{e}^{-(x\pm ib)}}\right| \mathrm{d}x \le 2\int_{\mathbb{R}^+} \frac{\mathrm{e}^{-tx\cos b}}{1+\mathrm{e}^{-x}}\,\mathrm{d}x \le 2\int_{\mathbb{R}^+} \mathrm{e}^{-tx\cos b}\,\mathrm{d}x \le 2(t\cos b)^{-1}$$

as well as

$$\int_{\mathbb{R}^-} \left|\frac{\mathrm{e}^{-t\ln(1+\mathrm{e}^{x\pm ib})}}{1+\mathrm{e}^{-(x\pm ib)}}\right| \mathrm{d}x \le 2\int_{\mathbb{R}^+} \frac{\mathrm{e}^{-\frac{t}{2}\mathrm{e}^{-x}\cos b}}{1+\mathrm{e}^x}\,\mathrm{d}x = 2\int_0^1 \frac{\mathrm{e}^{-\frac{t}{2}\xi\cos b}}{(1+\xi^{-1})\xi}\,\mathrm{d}\xi \le 4(t\cos b)^{-1},$$

where we have used the substitution $x = -\ln\xi$.

Applying ([29], Thm. 3.2.1), we thus obtain

$$\left|\frac{1}{t} - \sum_{k\in\mathbb{Z}} h\,w(kh)\mathrm{e}^{-\alpha(kh)\,t}\right| \le 12(t\cos b)^{-1}\frac{\mathrm{e}^{-\pi b/h}}{2\sinh(\pi b/h)} \le 24(t\cos b)^{-1}\mathrm{e}^{-2\pi b/h} \le \frac{1}{2}t^{-1}\delta$$

for the range of $h$ given in the assertion. Here we have used that in particular, $h \le 2\pi b/\ln 2$, which gives $\mathrm{e}^{-\pi b/h}/(2\sinh(\pi b/h)) \le 2\mathrm{e}^{-2\pi b/h}$, and that $\ln 48 < 4$.

The estimates for $n^+$ and $n$ follow from the decay of the integrand on $\mathbb{R}$: on the one hand, we have

$$\sum_{k>n^+} h\,w(kh)\mathrm{e}^{-\alpha(kh)\,t} \le h\int_{n^+}^\infty \mathrm{e}^{-txh}\,\mathrm{d}x \le t^{-1}\int_{n^+ht}^\infty \mathrm{e}^{-x}\,\mathrm{d}x \le t^{-1}\mathrm{e}^{-n^+h}.$$

The expression on the right hand side is bounded by $\frac{1}{2}t^{-1}\delta$ for $n^+ \geq h^{-1}(\ln 2 + |\ln \delta|)$, which yields (2.11). On the other hand,

$$\sum_{k<-n} h\, w(kh)\mathrm{e}^{-\alpha(kh)\, t} \leq \int_{nh}^{\infty} \mathrm{e}^{-x}\, \mathrm{d}x \leq \mathrm{e}^{-nh},$$

and the expression on the right hand side is bounded by $t^{-1}\eta$ for all $t \in [1, T]$ for $n \geq h^{-1}(|\ln \eta| + \ln T)$. □

**Remark 2.6.** The integral representation (2.14) was also used in [20], where bounds on the absolute error are obtained by symmetric truncation of sinc approximations. A direct use of these estimates to bound relative errors as required for our purposes, however, would lead to a substantially less favorable result, since an absolute error tolerance would need to shrink with increasing range $T$. We thus instead directly derive a relative error bound.

Moreover, note that a related but slightly different relative error bound, for approximation of $t^{-1}$ on $(0, 1]$, was derived for a different purpose in [6]. Compared to a direct application of the latter result, which would involve a rescaling of coefficients in dependence on $T$, our above bound has the advantage that while keeping the upper summation index $n^+$ fixed, we can realize arbitrarily good approximations to a scaling operator equivalent to $\mathbf{S}^{-2}$ by simply adding additional separable terms. Although it may be possible to alternatively derive a result very similar to Theorem 2.5 using the approach of [6], the required effort appears to be comparable to the above direct argument.

In other works, preconditioners for low-rank tensor methods for fixed discretizations of second-order problems have been proposed, for instance in [1, 5, 22, 24]. However, these have not been analyzed in their overall effect on the complexity of the solution process.

In what follows, we *fix* $\delta \in (0, 1)$ and $h$, $n^+$ as in Theorem 2.5. For the corresponding $\varphi_{h,n}$ and $\varphi_{h,\infty}$ we define

$$p_{n,\nu} := \omega_{\min}^{-2}\, \varphi_{h,n}\big((\omega_\nu/\omega_{\min})^2\big), \quad p_\nu := \lim_{n\to\infty} p_{n,\nu} = \omega_{\min}^{-2}\, \varphi_{h,\infty}\big((\omega_\nu/\omega_{\min})^2\big),$$

where $\omega_{\min} := \min_{\nu\in\nabla^d} \omega_\nu$. We then set

$$\mathbf{P} := \operatorname{diag}(p_\nu), \quad \mathbf{P}_n := \operatorname{diag}(p_{n,\nu}). \tag{2.15}$$

Theorem 2.5 states that $\mathbf{P}$, $\mathbf{P}_n$ have the properties

$$\|(\mathbf{P} - \mathbf{S}^{-2})\mathbf{S}^2\| \leq \delta, \qquad \|(\mathbf{P} - \mathbf{P}_n)\mathbf{S}^2\, \mathrm{R}_{\Lambda_T}\| \leq \eta \quad \text{for } n \geq \lceil h^{-1}(|\ln \eta| + \ln T)\rceil.$$

In other words, $\mathbf{P}$ is an approximation of $\mathbf{S}^{-2}$ with a *relative* error bound $\delta$, and $\mathbf{P}_n$ provides a finite-rank approximation to $\mathbf{P}$ for any prescribed relative error bound $\eta$ on compactly supported sequences. We shall use $\mathbf{P}$ which, in turn, is approximated by $\mathbf{P}_n$, as a substitute for $\mathbf{S}^{-2}$ in (2.8) when solving

$$\mathbf{T}\mathbf{u} = \mathbf{g}$$

by a Jacobi-type iteration. The modified idealized iteration thus has the form

$$\mathbf{u}_{j+1} = \mathbf{u}_j - \omega\mathbf{P}(\mathbf{T}\mathbf{u}_j - \mathbf{g}). \tag{2.16}$$

Setting $\mathbf{A} := \mathbf{P}\mathbf{T}$, $\mathbf{f} := \mathbf{P}\mathbf{g}$, this iteration will be realized in the perturbed form

$$\mathbf{u}_{j+1} = \mathbf{u}_j - \omega\mathbf{r}_j, \quad \mathbf{r}_j \approx (\mathbf{P}\mathbf{T})\mathbf{u}_j - \mathbf{P}\mathbf{g}$$

with a suitable approximation $\mathbf{r}_j$, involving $\mathbf{P}_n$, of the scaled residual.

## 3. Analysis of an adaptive method with error control in $L_2$

### 3.1. The adaptive scheme

The adaptive scheme to be proposed next has the following routines as main constituents:

- RECOMPRESS$(\mathbf{v};\eta)$, realizing the projection $\hat{\mathrm{P}}_\eta(\mathbf{v}) := \mathrm{P}_{\mathbb{U}(\mathbf{v}),\mathrm{r}(\mathbf{v},\eta)}\,\mathbf{v}$ from Section 1.1.2 with target accuracy $\eta$;
- COARSEN$(\mathbf{v};\eta)$, realizing the coarsening operator $\hat{\mathrm{C}}_\eta(\mathbf{v})$ from (1.13);
- RHS$(\eta)$, producing an $\eta$-accurate approximation to the right hand side $\mathbf{f}$;
- APPLY$(\mathbf{v};\eta)$, which yields $\mathbf{w}_\eta$ of finite support and ranks such that $\|\mathbf{A}\mathbf{v}-\mathbf{w}_\eta\|\le\eta$.

For a discussion of the first three routines we refer to [3, 4] and defer the precise description of APPLY$(\mathbf{v};\eta)$ to Section 3.3. We formulate the perturbed version of the idealized iteration (2.16) in Algorithm 1.

**Algorithm 1** $\mathbf{u}_\varepsilon = \mathrm{SOLVE}(\mathbf{A},\mathbf{f};\varepsilon)$

input $\begin{cases}\omega>0 \text{ and } \rho\in(0,1) \text{ such that } \|\mathrm{id}-\omega\mathbf{P}^{\frac12}\mathbf{T}\mathbf{P}^{\frac12}\|\le\rho,\\ c_{\mathbf{A}}\ge\|\mathbf{A}^{-1}\|,\ \varepsilon_0\ge c_{\mathbf{A}}\|\mathbf{f}\|,\\ \kappa_1,\kappa_2,\kappa_3\in(0,1) \text{ with } \kappa_1+\kappa_2+\kappa_3\le1, \text{ and } \beta_1\ge0,\ \beta_2>0.\end{cases}$

output $\mathbf{u}_\varepsilon$ satisfying $\|\mathbf{u}_\varepsilon-\mathbf{u}\|\le\varepsilon$.

$\mathbf{u}_0:=0$, $k:=0$
while $2^{-k}\varepsilon_0>\varepsilon$
  $\eta_{k,0}:=\rho 2^{-k}\varepsilon_0$
  $\mathbf{w}_{k,0}:=\mathbf{u}_k$
  $\mathbf{r}_{k,0}:=\mathrm{APPLY}(\mathbf{w}_{k,0};\frac12\eta_{k,0})-\mathrm{RHS}(\frac12\eta_{k,0})$
  $j\leftarrow0$
  while $c_{\mathbf{A}}(\|\mathbf{r}_{k,j}\|+\eta_{k,j})>\kappa_1 2^{-(k+1)}\varepsilon_0$
    $\mathbf{w}_{k,j+1}:=\mathrm{COARSEN}\big(\mathrm{RECOMPRESS}(\mathbf{w}_{k,j}-\omega\mathbf{r}_{k,j};\beta_1\eta_{k,j});\beta_2\eta_{k,j}\big)$
    $j\leftarrow j+1$.
    $\eta_{k,j}:=\rho^{j+1}2^{-k}\varepsilon_0$
    $\mathbf{r}_{k,j}:=\mathrm{APPLY}(\mathbf{w}_{k,j};\frac12\eta_{k,j})-\mathrm{RHS}(\frac12\eta_{k,j})$
  end while
  $\mathbf{u}_{k+1}:=\mathrm{COARSEN}\big(\mathrm{RECOMPRESS}(\mathbf{w}_{k,j};\kappa_2 2^{-(k+1)}\varepsilon_0);\kappa_3 2^{-(k+1)}\varepsilon_0\big)$
  $k\leftarrow k+1$
end while
$\mathbf{u}_\varepsilon:=\mathbf{u}_k$

### 3.2. Convergence analysis

We address first the convergence of the idealized iteration (2.16).

**Remark 3.1.** Since $\Omega$ is bounded, $A$ has a purely discrete spectrum and all eigenfunctions of $A$ belong to $\mathrm{H}^1_0(\Omega)$. As a consequence, $\mathbf{A}=\mathbf{P}\mathbf{T}$ and $\mathbf{P}^{\frac12}\mathbf{T}\mathbf{P}^{\frac12}$ have the same spectrum, where we recall that $\mathbf{P}^{\frac12}$ is spectrally equivalent to $\mathbf{S}^{-1}$.

Let $\omega>0$ be chosen such that $\rho:=\|\mathrm{id}-\omega\mathbf{P}^{\frac12}\mathbf{T}\mathbf{P}^{\frac12}\|<1$. Since the eigenvalues of $\mathrm{id}-\omega\mathbf{P}^{\frac12}\mathbf{T}\mathbf{P}^{\frac12}$ and $\mathbf{C}:=\mathrm{id}-\omega\mathbf{A}$ coincide, we have

$$\lim_{k\to\infty}\|\mathbf{C}^k\|^{\frac1k}=\rho. \tag{3.1}$$

Consequently, for an arbitrarily fixed $\tilde\rho$ with $\rho<\tilde\rho<1$, this implies the following: there exist $K\in\mathbb{N}$ and $B>0$ such that

$$\|\mathbf{C}^k\|\le\tilde\rho^k \quad \text{for } k>K=K(\tilde\rho), \qquad \|\mathbf{C}^k\|\le B=B(\tilde\rho) \quad \text{for } k\le K, \tag{3.2}$$

which confirms the convergence of (2.16). It now remains to account for the additional perturbations in Algorithm 1.

**Proposition 3.2.** *For any given target accuracy $\varepsilon > 0$, Algorithm 1 terminates after finitely many steps and yields a finitely supported tensor $\mathbf{u}_\varepsilon$, satisfying*

$$\|u - u_\varepsilon\|_{\mathrm{L}_2(\Omega)} = \|\mathbf{u} - \mathbf{u}_\varepsilon\| \le \varepsilon, \tag{3.3}$$

*where $u$ is the exact solution of (1.1), whose $\mathrm{L}_2$-wavelet coefficient array $\mathbf{u}$ satisfies (1.7), and $\mathbf{u}_\varepsilon$ is the coefficient tensor of $u_\varepsilon$.*

*Proof.* The argument is similar to that in [4] and differs only in the treatment of the inner loop between steps 7 and 12 in Algorithm 1. For convenience we briefly sketch the induction argument that shows that $\|\mathbf{u}_k - \mathbf{u}\| \le 2^{-k}\varepsilon_0$. To that end, since by step 5,

$$\|\mathbf{w}_{k,j} - \mathbf{u}\| \le \|\mathbf{A}^{-1}\|\|\mathbf{A}\mathbf{w}_{k,j} - \mathbf{f}\| \le c_{\mathbf{A}}(\|\mathbf{r}_{k,j}\| + \eta_{k,j}),$$

condition 7 ensures that when exiting the inner loop at step 12, the approximation $\mathbf{w}_{k,j}$ satisfies $\|\mathbf{w}_{k,j} - \mathbf{u}\| \le \kappa_1 2^{-(k+1)}\varepsilon_0$. To see that $c_{\mathbf{A}}(\|\mathbf{r}_{k,j}\| + \eta_{k,j})$ indeed becomes as small as one wishes when $j$ increases, one derives from steps 5, 8, and the definition of $\eta_{k,j}$ in step 10, that the iterates $\mathbf{w}_{k,j}$ satisfy a relation of the form

$$\mathbf{w}_{k,j+1} = \mathbf{w}_{k,j} - \omega\mathbf{A}\mathbf{w}_{k,j} + \omega\mathbf{f} + \mathbf{z}_{k,j},$$

where $\|\mathbf{z}_{k,j}\| \le (\beta_1 + \beta_2 + \omega)\eta_{k,j} =: \varepsilon_{k,j}$. Using $\mathbf{w}_{k,j+1} - \mathbf{u} = \mathbf{C}(\mathbf{w}_{k,j} - \mathbf{u}) + \mathbf{z}_{k,j}$, we thus obtain, for $j > K$,

$$\begin{aligned}\|\mathbf{w}_{k,j} - \mathbf{u}\| &\le \|\mathbf{C}^j\|\|\mathbf{w}_{k,0} - \mathbf{u}\| + \sum_{\ell=0}^{j-1}\|\mathbf{C}^{j-1-\ell}\|\|\mathbf{z}_{k,\ell}\| \\ &\le \tilde\rho^j\|\mathbf{w}_{k,0} - \mathbf{u}\| + \sum_{\ell=0}^{j-1-K}\tilde\rho^{j-1-\ell}\|\mathbf{z}_{k,\ell}\| + B\sum_{\ell=j-K}^{j-1}\|\mathbf{z}_{k,\ell}\|.\end{aligned}$$

Since $\|\mathbf{z}_{k,\ell}\| \le \varepsilon_{k,\ell} \le (\beta_1 + \beta_2 + \omega)\rho^\ell 2^{-k}\varepsilon_0$ we conclude that for $\beta_3 := (\beta_1 + \beta_2 + \omega)$,

$$\begin{aligned}\|\mathbf{w}_{k,j} - \mathbf{u}\| &\le \tilde\rho^j\|\mathbf{u}_k - \mathbf{u}\| + \big((j-K)\tilde\rho^{j-1} \\ &\qquad + (1-\rho)^{-1}(\rho^{-K} - 1)B\rho^j\big)\beta_3 2^{-k}\varepsilon_0 \\ &\le \big\{\tilde\rho^j + \big((j-K)\tilde\rho^{j-1} + (1-\rho)^{-1}(\rho^{-K} - 1)B\rho^j\big)\beta_3\big\}2^{-k}\varepsilon_0.\end{aligned} \tag{3.4}$$

On the other hand, observing that

$$\|\mathbf{r}_{k,j}\| \le \|\mathbf{A}\mathbf{w}_{k,j} - \mathbf{f}\| + \eta_{k,j} \le \|\mathbf{A}\|\|\mathbf{w}_{k,j} - \mathbf{u}\| + \eta_{k,j},$$

we see that after at most a finite number $J$ of steps, depending only on $\mathbf{A}$ (*i.e.*, on the operator $A$ and the chosen wavelet basis), indeed $c_{\mathbf{A}}(\|\mathbf{r}_{k,J}\| + \eta_{k,J}) \le \kappa_1 2^{-(k+1)}\varepsilon_0$ holds, the inner loop terminates and hence $\|\mathbf{w}_{k,J} - \mathbf{u}\| \le \kappa_1 2^{-(k+1)}\varepsilon_0$. For later reference, note that $J \le I$ with

$$I := \min\Big\{j \ge K\colon c_{\mathbf{A}}\Big(\|\mathbf{A}\|\big[\tilde\rho^j + \big((j-K)\tilde\rho^{j-1} + (1-\rho)^{-1}(\rho^{-K} - 1)B\rho^j\big)\beta_3\big] + 2\rho^{j+1}\Big) \le \frac{\kappa_1}{2}\Big\}. \tag{3.5}$$

Since $\kappa_1 + \kappa_2 + \kappa_3 \le 1$, we obtain $\|\mathbf{u}_{k+1} - \mathbf{u}\| \le 2^{-(k+1)}\varepsilon$. □

### 3.3. Operator approximation

Our approximate application of the high-dimensional operator $\mathbf{A}$ is based on the wavelet compressibility properties of the one-dimensional operators

$$\mathbf{A}_2^{(i)} := \hat{\mathbf{S}}_i^{-2}\mathbf{T}_2, \quad \mathbf{A}_3^{(i)} := \hat{\mathbf{S}}_i^{-1}\mathbf{T}_3 = -\hat{\mathbf{S}}_i^{-1}\mathbf{T}_4, \tag{3.6}$$

where the last relation holds because of (1.10) and the boundary conditions. More precisely, we make use of the following property: there exist an $s > 0$ and $\mathbf{T}_{n,j}$, $j \in \mathbb{N}$, such that for some fixed sequences of positive numbers $\beta(\mathbf{A}_n^{(i)}) \in \ell_1$ for $n = 2, 3$,

$$\|\hat{\mathbf{S}}_i^{-2}(\mathbf{T}_2 - \mathbf{T}_{2,j})\| \le \beta_j(\mathbf{A}_2^{(i)})\, 2^{-sj}, \quad \|\hat{\mathbf{S}}_i^{-1}(\mathbf{T}_3 - \mathbf{T}_{3,j})\| \le \beta_j(\mathbf{A}_3^{(i)})\, 2^{-sj}, \tag{3.7}$$

where each $\mathbf{T}_{n,j}$ has at most $\alpha_j(\mathbf{A}_n^{(i)})\, 2^j$ nonzero entries in each column, where $\alpha(\mathbf{A}_n^{(i)}) \in \ell_1$ are additional fixed sequences of positive numbers. It is convenient to scale the sequences so that $\|\beta(\mathbf{A}_n^{(i)})\|_{\ell_1} \le \|\mathbf{A}_n^{(i)}\|$.

Note that this is slightly weaker than the usual definition of $s^*$-compressibility [9], since we do not require a bound on the number of entries per row, and we shall refer to the property in (3.7) as *column-$s^*$-compressibility*. In addition, as in [3] we assume the approximations to have the *level decay property*, that is, there exists a $\gamma > 0$ such that $||\nu| - |\mu|| > \gamma j$ implies $T_{n,j,\nu\mu} = 0$.

Our aim is to obtain $\mathbf{w}_\eta$, satisfying certain representation complexity bounds, such that $\|\mathbf{P}\mathbf{T}\mathbf{v} - \mathbf{w}_\eta\| \le \eta$. We make the ansatz $\mathbf{w}_\eta = \mathbf{P}_n\tilde{\mathbf{T}}\mathbf{v}$ where $\mathbf{P}_n$ is the finite rank approximation to the scaling operator $\mathbf{P}$ from (2.15) and $\tilde{\mathbf{T}}$ is a "compressed" version of $\mathbf{T}$. Specifically, based on the estimate

$$\begin{aligned}\|\mathbf{P}\mathbf{T}\mathbf{v} - \mathbf{P}_n\tilde{\mathbf{T}}\mathbf{v}\| &\le \|\mathbf{P}(\mathbf{T} - \tilde{\mathbf{T}})\mathbf{v}\| + \|(\mathbf{P} - \mathbf{P}_n)\tilde{\mathbf{T}}\mathbf{v}\| \\ &\le (1+\delta)\|\mathbf{S}^{-2}(\mathbf{T} - \tilde{\mathbf{T}})\mathbf{v}\| + \|(\mathbf{P} - \mathbf{P}_n)\tilde{\mathbf{T}}\mathbf{v}\|,\end{aligned} \tag{3.8}$$

we first choose $\tilde{\mathbf{T}} = \tilde{\mathbf{T}}(\mathbf{v})$ depending on $\mathbf{v}$ to obtain a suitable bound on the first term on the right hand side, and subsequently pick $n$ such that the second term is sufficiently small.

The construction of $\tilde{\mathbf{T}}$, based on the property (3.7), can be done in complete analogy to ([3], Sect. 4.2). The resulting approximation is of the form

$$\tilde{\mathbf{T}} = \tilde{\mathbf{T}}_J := \sum_{\mathsf{n}\in\mathsf{K}_d(\mathsf{R})} c_{\mathsf{n}} \bigotimes_i \tilde{\mathbf{T}}_{n_i}^{(i)},$$

where $\tilde{\mathbf{T}}_1^{(1)} = \mathbf{T}_1 = \mathrm{id}$ and for $n_i > 1$,

$$\tilde{\mathbf{T}}_{n_i}^{(i)} = \tilde{\mathbf{T}}_{n_i}^{(i,J)} := \sum_{p=0}^{J+1} \mathbf{T}_{n_i,[p]}^{(i,J)}\, \mathrm{R}_{\Lambda_{[p]}^{(i)}} \tag{3.9}$$

with $\mathbf{T}_{n_i,[p]}^{(i)} := \mathbf{T}_{n_i,J-p}$, $p = 0, \dots, J$, and $\mathbf{T}_{n_i,[J+1]}^{(i)} := 0$ as in (3.7). Recall from Section 1.1.2 that the operator $\mathrm{R}_\Lambda$ retains the entries of a tensor supported in $\Lambda$ and replaces all others by zero. The adaptive $\mathbf{v}$-dependent formation of $\tilde{\mathbf{T}}$ hinges on the choice of the index sets $\Lambda_{[p]}^{(i)}$, which are constructed from the supports $\bar{\Lambda}_j^{(i)}$ of the best $2^j$-term approximations of $\pi^{(i)}(\mathbf{v})$. Specifically, setting $\bar{\Lambda}_{-1}^{(i)} := \emptyset$, we recursively define

$$\Lambda_{[p]}^{(i)} := \bar{\Lambda}_p^{(i)} \setminus \bar{\Lambda}_{p-1}^{(i)},\ p = 0, \dots, J, \quad \Lambda_{[J+1]}^{(i)} := \nabla \setminus \bar{\Lambda}_J^{(i)}, \quad \Lambda_{[p]}^{(i)} := \emptyset,\ p > J+1.$$

Defining next the *a posteriori* error indicator

$$e_J(\mathbf{v}) := \sum_{i=1}^d C_{\mathbf{A}}^{(i)} \left[\sum_{p=0}^J \left(\sum_{n=2}^R \beta_{J-p}(\mathbf{A}_n^{(i)})\right) 2^{-s(J-p)} \|\mathrm{R}_{\Lambda_{[p]}^{(i)}}\pi^{(i)}(\mathbf{v})\| + \sum_{n=2}^R \|\mathbf{A}_n^{(i)}\|\, \|\mathrm{R}_{\Lambda_{[J+1]}^{(i)}}\pi^{(i)}(\mathbf{v})\|\right], \tag{3.10}$$

where

$$C_{\mathbf{A}}^{(i)} := \max\left\{|a_{ii}|, 2\sum_{j\neq i}\|\mathbf{A}_3^{(j)}\||a_{ij}|,\right\} \leq \max\left\{1, 2\max_{j\neq i}\|\mathbf{A}_3^{(j)}\|\right\}|a_{ii}|, \tag{3.11}$$

one can follow the arguments in ([3], Lem. 6.10), now using (3.7), to verify that

$$\|\mathbf{S}^{-2}(\mathbf{T}-\tilde{\mathbf{T}}_J)\mathbf{v}\| \leq e_J(\mathbf{v}). \tag{3.12}$$

The heart of Algorithm 1 is the adaptive application of $\mathbf{A}$. We can now specify the corresponding routine $\text{APPLY}(\mathbf{v};\eta)$ for a finitely supported input $\mathbf{v}\in\ell_2(\nabla^d)$ and a prescribed error tolerance $\eta>0$. The relevant properties are collected in the following theorem, which is a complete analog to Theorem 6.8 in [3].

Without loss of generality, for a given $\mathbf{v}$ we shall employ tolerances $\eta \leq \|\mathbf{S}^{-2}\mathbf{T}\|\|\mathbf{v}\|$, since otherwise we may choose $\mathbf{w}_\eta = 0$. For such $\eta$, it will be convenient to define

$$\zeta(\eta;\mathbf{v}) := \frac{\eta}{3\|\mathbf{S}^{-2}\mathbf{T}\|\|\mathbf{v}\|}. \tag{3.13}$$

**Theorem 3.3.** *Given any $\mathbf{v}\in\ell_2(\nabla^d)$ of finite support and finite hierarchical ranks as well as any $0<\eta\leq\|\mathbf{S}^{-2}\mathbf{T}\|\|\mathbf{v}\|$, let $\mathbf{w}_\eta$ be defined as follows: choose $J(\eta;\mathbf{v})$ as the minimal integer such that*

$$(1+\delta)e_{J(\eta;\mathbf{v})}(\mathbf{v}) \leq \frac{\eta}{2}, \tag{3.14}$$

*and set $\mathbf{w}_\eta := \mathbf{P}_{m(\eta;\mathbf{v})}\tilde{\mathbf{T}}_{J(\eta;\mathbf{v})}\mathbf{v}$ where, with $\tilde{\Lambda} := \times_{i=1}^d \operatorname{supp}_i(\tilde{\mathbf{T}}_{J(\eta;\mathbf{v})}\mathbf{v})$,*

$$m(\eta;\mathbf{v}) := \left\lceil h^{-1}\left(|\ln(\zeta(\eta;\mathbf{v})| + \ln\max\{(\omega_\nu/\omega_{\min})^2 : \nu\in\tilde{\Lambda}\}\right)\right\rceil. \tag{3.15}$$

*Then the following statements hold:*

(i) *We have the estimates*

$$\|\mathbf{A}\mathbf{v}-\mathbf{w}_\eta\| \leq \eta, \tag{3.16}$$

$$\#\operatorname{supp}_i(\mathbf{w}_\eta) \leq \|\hat{\alpha}\|_{\ell_1}\eta^{-\frac{1}{s}}\left(4(2^s+2)R^{1+s}\sum_{i=1}^d C_{\mathbf{A}}^{(i)}\max_{n>1}\|\mathbf{A}_n^{(i)}\|\,\|\pi^{(i)}(\mathbf{v})\|_{\mathcal{A}^s}\right)^{\frac{1}{s}}, \tag{3.17}$$

*where $\hat{\alpha} := (\hat{\alpha}_k)_{k\in\mathbb{N}}$ and $\hat{\alpha}_k := \max_{i\in\{1,\dots,d\}}\max_{n>1}\alpha_k(\mathbf{A}_n^{(i)})$.*

(ii) *The outputs of* APPLY *are sparsity-stable in the sense that for $i\in\{1,\dots,d\}$,*

$$\|\pi^{(i)}(\mathbf{w}_\eta)\|_{\mathcal{A}^s} \leq \left(\check{C}_{\mathbf{A}}^{(i)} + \frac{2^{3s+2}}{2^s-1}\|\hat{\alpha}\|_{\ell_1}^s\max_{n>1}\|\mathbf{A}_n^{(i)}\|\,C_{\mathbf{A}}^{(i)}\right)R^s(1+\delta)\,\|\pi^{(i)}(\mathbf{v})\|_{\mathcal{A}^s}, \tag{3.18}$$

*where $C_{\mathbf{A}}^{(i)}$ is defined in (3.11) and*

$$\check{C}_{\mathbf{A}}^{(i)} := 12\,(d-1)\max_{j\neq i}|a_{jj}|\left(\max_{i,n_i}\|\mathbf{A}_{n_i}^{(i)}\|\right)^2. \tag{3.19}$$

(iii) *Denoting by $R_\alpha$ the hierarchical ranks in the representation of $\mathbf{T}$, the hierarchical ranks of $\mathbf{w}_\eta$ can be bounded by*

$$\operatorname{rank}_\alpha(\mathbf{w}_\eta) \leq \hat{m}(\eta;\mathbf{v})R_\alpha\operatorname{rank}_\alpha(\mathbf{v}), \quad \alpha\in\mathcal{D}_d, \tag{3.20}$$

*where for $n^+ = n^+(\delta)$ from in Section 2.2 and $m(\eta;\mathbf{v})$ defined in (3.15),*

$$\hat{m}(\eta;\mathbf{v}) := 1 + n^+ + m(\eta;\mathbf{v}). \tag{3.21}$$

(iv) *The number* $\mathrm{ops}(\mathbf{w}_\eta)$ *of floating point operations required to compute* $\mathbf{w}_\eta$ *in the hierarchical Tucker format for a given* $\mathbf{v}$ *with ranks* $\mathrm{rank}_\alpha(\mathbf{v}) = r_\alpha$, $\alpha \in \mathcal{D}_d \setminus \{0_d\}$, *and* $r_{0_d} = 1$, *scales like*

$$\mathrm{ops}(\mathbf{w}_\eta) \lesssim \sum_{\alpha\in\mathcal{N}(\mathcal{D}_d)} \big(\hat m(\eta;\mathbf{v})\big)^3 R_\alpha r_\alpha \prod_{q=1}^{2} R_{c_q(\alpha)} r_{c_q(\alpha)} + \eta^{-1/s} \sum_{i=1}^{d} \|\hat\alpha\|_{\ell_1} \hat m(\eta;\mathbf{v}) R r_i \left( \sum_{j=1}^{d} C_{\mathbf{A}}^{(j)} R \|\pi^{(j)}(\mathbf{v})\|_{\mathcal{A}^s} \right)^{1/s}, \tag{3.22}$$

*where the constant is independent of* $\eta, \mathbf{v}$, *and* $d$.

(v) *Assume in addition that the approximations* $\mathbf{T}_{n,j}$ *have the level decay property. With the notation* $L(\mathbf{v}) := \max\{|\nu_i| \colon \nu_i \in \mathrm{supp}_i(\mathbf{v}),\ i = 1,\dots,d\}$, *the scaling ranks* $\hat m(\eta;\mathbf{v})$, *defined in* (3.21), *can be bounded by*

$$\hat m(\eta;\mathbf{v}) \le C(\delta, s, \mathbf{A}) \left[ 1 + L(\mathbf{v}) + |\ln \eta| + \ln \left( \sum_{i=1}^{d} \|\pi^{(i)}(\mathbf{v})\|_{\mathcal{A}^s} \right) \right]. \tag{3.23}$$

**Remark 3.4.** Comparing the above statements with Theorem 6.8 in [3] reveals several minor differences. This concerns, for instance, the more favorable constants in (3.17). Moreover, the condition (3.14) is slightly relaxed here and, due to the one-sided application of the scaling operator, the definition of $m(\eta;\mathbf{v})$ in (3.15) is somewhat less involved. The main difference lies in the rank bounds (3.20) and in the bound on the number of operations (3.22), where $\hat m(\eta;\mathbf{v})$ enters with half the exponent of ([3], Thm. 6.8).

The proof of Theorem 3.3 differs from the proof of Theorem 6.8 in [3] only in minor technical details. In fact, the one-sided scaling simplifies some of the arguments. We therefore give some brief comments and omit a complete proof.

First, with $\tilde\Lambda$ as in Theorem 3.3, one has

$$\|(\mathbf{P} - \mathbf{P}_{m(\eta;\mathbf{v})})\tilde{\mathbf{T}}\mathbf{v}\| \le \|(\mathbf{P} - \mathbf{P}_{m(\eta;\mathbf{v})})\mathbf{S}^2 \,\mathrm{R}_{\tilde\Lambda}\| \big(e_{J(\eta;\mathbf{v})} + \|\mathbf{S}^{-1}\mathbf{T}\|\|\mathbf{v}\|\big).$$

Combining this with (3.8), (3.12), and (3.15) yields

$$\|\mathbf{A}\mathbf{v} - \mathbf{w}_\eta\| \le (1+\delta) e_{J(\eta;\mathbf{v})}(\mathbf{v}) + \zeta(\eta;\mathbf{v}) \big(e_{J(\eta;\mathbf{v})}(\mathbf{v}) + \|\mathbf{S}^{-2}\mathbf{T}\|\|\mathbf{v}\|\big).$$

In view of (3.14), $\eta \le \|\mathbf{S}^{-2}\mathbf{T}\|\|\mathbf{v}\|$, and (3.13), this confirms (3.16). The argument for (3.17) is the same as in [3]. The slightly different constant results from the relaxed requirement (3.14) on $J(\eta;\mathbf{v})$. The appearance of the factor $(1+\delta)$ in (3.18) instead of $(1+\delta)^2$ in ([3], Thm. 6.8) results again from the one-sided scaling, which also leads to the more favorable exponents in (3.20) and (3.22).

### 3.4. Complexity estimates

We have seen that Algorithm 1 converges without any specific assumptions on the solution in the sense that a given target accuracy is reached after finitely many steps. We will show next that, under canonical assumptions on the problem data $(\mathbf{A}, \mathbf{f})$, whenever the solution has certain sparsity properties (regarding low-rank approximability and representations sparsity of the tensor factors), the approximate solution produced by Algorithm 1 has similar and in a sense near-optimal sparsity properties. We proceed now formulating our data assumptions as well as the envisaged *benchmark assumptions* concerning the solution. These assumptions are not required for ensuring the convergence of the algorithm, which holds in the more general setting of Proposition 3.2. It also needs to be emphasized that these assumptions are *not* explicitly used by the algorithm, but rather exploited automatically.

From the results in [3] and Theorem 2.1, we know that the infinite matrices $\mathbf{S}^{-2}\mathbf{T}$ and $\mathbf{S}^{-1}\mathbf{T}\mathbf{S}^{-1}$ are automorphisms of $\ell_2(\nabla^d)$. In particular, $\hat{\mathbf{S}}_i^{-2}\mathbf{T}_2$ and $\hat{\mathbf{S}}_i^{-1}\mathbf{T}_2\hat{\mathbf{S}}_i^{-1}$ are bounded mappings on $\ell_2(\nabla)$. This latter fact can be interpreted as follows. Let $\ell_2^t(\nabla)$ denote the *weighted* space $\{\mathbf{w} \in \mathbb{R}^\nabla : \|\hat{\mathbf{S}}^t\mathbf{w}\| < \infty\}$, which defines a scale of interpolation spaces. Then, the boundedness of $\hat{\mathbf{S}}_i^{-1}\mathbf{T}_2\hat{\mathbf{S}}_i^{-1}$ means that $\hat{\mathbf{S}}_i^{-2}\mathbf{T}_2 : \ell_2^1(\nabla) \to \ell_2^1(\nabla)$ is bounded.

By interpolation, $\hat{\mathbf{S}}_i^{-2}\mathbf{T}_2 : \ell_2^t(\nabla) \to \ell_2^t(\nabla)$ is bounded for $t \in [0,1]$. This, in turn, means that $\hat{\mathbf{S}}_i^{t-2}\mathbf{T}_2\hat{\mathbf{S}}_i^{-t}$ is bounded for $t \in [0,1]$, and by the same argument we obtain also that $\hat{\mathbf{S}}_i^{t-1}\mathbf{T}_3\hat{\mathbf{S}}_i^{-t}$ is bounded. Hence, for $t \in [0,1]$,

$$\|\hat{\mathbf{S}}_i^t\mathbf{A}_2^{(i)}\hat{\mathbf{S}}_i^{-t}\|,\ \|\hat{\mathbf{S}}_i^t\mathbf{A}_3^{(i)}\hat{\mathbf{S}}_i^{-t}\| < \infty \quad \text{as well as} \quad \|\mathbf{S}^t\mathbf{f}\| \le (1+\delta)\|\mathbf{S}^{t-2}\mathbf{g}\| < \infty. \tag{3.24}$$

The *excess regularity* assumption made in [3] corresponds to the statement that (3.24) holds for *some* $t > 0$, which there indeed had to be assumed. As shown by the above considerations, however, this is in our present setting automatically satisfied for $t = 1$.

We now formulate our data assumptions.

**Assumption 3.5.** Concerning the scaled matrix representation $\mathbf{A}$ and the right hand side $\mathbf{f}$ we require the following properties for some fixed $s^* > 0$:

(i) The lower-dimensional component operators $\mathbf{A}_{ni}^{(i)}$, defined in (3.6), are column-$s^*$-compressible with the level decay property (*cf.* Sect. 3.3).
(ii) The number of operations required for evaluating each entry in the approximations $\mathbf{T}_{n,j}$ as in (3.7) is uniformly bounded.
(iii) We have an estimate $c_{\mathbf{A}} \ge \|\mathbf{A}^{-1}\|$, and the initial error estimate $\varepsilon_0$ overestimates the true value of $\|\mathbf{A}^{-1}\|\|\mathbf{f}\|$ only up to some absolute multiplicative constant, *i.e.*, $\varepsilon_0 \lesssim \|\mathbf{A}^{-1}\|\|\mathbf{f}\|$.
(iv) The contractions of $\mathbf{f}$ are compressible, *i.e.*, $\pi^{(i)}(\mathbf{f}) \in \mathcal{A}^s$, $i = 1,\dots,d$, for any $s$ with $0 < s < s^*$.

The concrete realization of the routine RHS depends on the concrete way the right hand side is given. For details on possible constructions of RHS, we refer to ([3], Appendix B), which justifies the following assumptions made in subsequent complexity statements.

**Assumption 3.6.** The procedure RHS is assumed to have the following properties:

(v) There exists an approximation $\mathbf{f}_\eta := \mathrm{RHS}(\eta)$ such that $\|\mathbf{f} - \mathrm{RHS}(\eta)\| \le \eta$ and the following inequalities hold:

$$\|\pi^{(i)}(\mathbf{f}_\eta)\|_{\mathcal{A}^s} \le C^{\mathrm{sparse}}\|\pi^{(i)}(\mathbf{f})\|_{\mathcal{A}^s}, \quad \|\mathbf{S}_i\mathbf{f}_\eta\| \le C^{\mathrm{reg}}\|\mathbf{S}_i\mathbf{f}\|,$$
$$\sum_i \#\operatorname{supp}_i(\mathbf{f}_\eta) \le C^{\mathrm{supp}}\, d\, \eta^{-\frac{1}{s}} \left(\sum_i \|\pi^{(i)}(\mathbf{f})\|_{\mathcal{A}^s}\right)^{\frac{1}{s}},$$
$$|\operatorname{rank}(\mathbf{f}_\eta)|_\infty \le C_{\mathbf{f}}^{\mathrm{rank}}\,|\ln\eta|^{b_{\mathbf{f}}}.$$

Here the constants $C^{\mathrm{sparse}}, C^{\mathrm{supp}}, C^{\mathrm{reg}}, C_{\mathbf{f}}^{\mathrm{rank}} > 0$, $b_{\mathbf{f}} \ge 1$, are independent of $\eta$, and $C^{\mathrm{sparse}}$, $C^{\mathrm{reg}}$, $C^{\mathrm{supp}}$ are independent of $\mathbf{f}$.
(vi) The number of operations required for evaluating $\mathrm{RHS}(\eta)$ is bounded, with a constant $C_{\mathbf{f}}^{\mathrm{ops}}(d)$, by $\mathrm{ops}(\mathbf{f}_\eta) \le C_{\mathbf{f}}^{\mathrm{ops}}(d)\big[|\ln\eta|^{3b_{\mathbf{f}}} + |\ln\eta|^{b_{\mathbf{f}}}\eta^{-\frac{1}{s}}\big]$.

Next we explain the benchmark properties of the solution to which subsequent complexity statements refer. These properties are *not* used by the solver.

**Assumption 3.7.** Concerning the approximability of the solution $\mathbf{u}$, we assume:

(vii) $\mathbf{u} \in \mathcal{A}_{\mathcal{H}}(\gamma_{\mathbf{u}})$ with $\gamma_{\mathbf{u}}(n) = \mathrm{e}^{d_{\mathbf{u}}n^{1/b_{\mathbf{u}}}}$ for some $d_{\mathbf{u}} > 0$, $b_{\mathbf{u}} \ge 1$.
(viii) $\pi^{(i)}(\mathbf{u}) \in \mathcal{A}^s$ for $i = 1,\dots,d$, for any $s$ with $0 < s < s^*$.

When discussing tractability issues in the sense of complexity theory it is important to know how the data behave with respect to the spatial dimension $d$.

**Assumption 3.8.** In our comparison of problems for different values of $d$, we assume:

(ix) The following constants are independent of $d$: $d_{\mathbf{u}}$, $b_{\mathbf{u}}$, $C^{\mathrm{sparse}}$, $C^{\mathrm{supp}}$, $C^{\mathrm{reg}}$, $C^{\mathrm{rank}}_{\mathbf{f}}$.
(x) The following quantities remain bounded independently of $d$: $\|\mathbf{A}\|$, $\|\mathbf{A}^{-1}\|$; the maximum hierarchical representation rank $\max_\alpha R_\alpha$ of $\mathbf{T}$; the quantities $\|\pi^{(i)}(\mathbf{u})\|_{\mathcal{A}^s}$ in the benchmark assumptions, $\|\pi^{(i)}(\mathbf{f})\|_{\mathcal{A}^s}$ in Assumptions 3.7(v), each for $i = 1, \dots, d$.
(xi) In addition, we assume that $C^{\mathrm{ops}}_{\mathbf{f}}(d)$ as in Assumptions 3.7(vi) grows at most polynomially as $d \to \infty$.
(xii) There exists a choice of $\tilde{\rho}$ in (3.2) independent of $d$ such that the corresponding values $K(\tilde{\rho})$, $B(\tilde{\rho})$ are bounded independently of $d$ as well.

The assumed $d$-independence of the parameters in the low-rank approximability of $\mathbf{u}$ in (ix) has been established in [11] for diagonal $M$. The numerical results in Section 4.2.2 support the conjecture that this still holds for certain tridiagonal $M$ with $d$-independent condition number. Concerning the assumptions on $\|\mathbf{A}\|$, $\|\mathbf{A}^{-1}\|$, see Theorem 2.1 in Section 2.1. Concerning (xii), we know from the discussion in Section 3.2 that the existence of $K$, $B$ as in (3.2) is ensured for $\tilde{\rho} > \rho$. Since the values of corresponding $K$ and $B$ are not explicitly quantified, however, the *a priori* bound on the number of steps (3.5) serves only theoretical purposes, and we have to rely on an *a posteriori* condition on the approximate residual for controlling the iteration. The concrete resulting values of $K$ and $B$ may depend on the choice of basis functions. We do not have a proof that these constants remain bounded independently of $d$ for our examples considered in Section 4. In the numerical results given there, however, we find that these values do not have any significant influence in practice: in the cases with $d$-independent $\|\mathbf{A}\|\|\mathbf{A}^{-1}\|$ (and hence $d$-independent $\rho$) we do not observe any deterioration of the convergence in the initial phase of each inner loop as $d$ is increased. This leads us to the conjecture that $K$ and $B$ then indeed remain bounded independently of $d$ as well.

The main result of this paper reads as follows.

**Theorem 3.9.** *Suppose that Assumptions 3.5, 3.6 hold and that Assumption 3.7 are valid for the solution* $\mathbf{u}$ *of* $\mathbf{A}\mathbf{u} = \mathbf{f}$. *Let* $\alpha > 0$ *and let* $\kappa_{\mathrm{P}}, \kappa_{\mathrm{C}}$ *be as in Theorem 1.1. Let the constants* $\kappa_1, \kappa_2, \kappa_3$ *in Algorithm 1 be chosen as*

$$\kappa_1 = \big(1 + (1+\alpha)(\kappa_{\mathrm{P}} + \kappa_{\mathrm{C}} + \kappa_{\mathrm{P}}\kappa_{\mathrm{C}})\big)^{-1},$$
$$\kappa_2 = (1+\alpha)\kappa_{\mathrm{P}}\kappa_1, \qquad \kappa_3 = \kappa_{\mathrm{C}}(\kappa_{\mathrm{P}}+1)(1+\alpha)\kappa_1,$$

*and let* $\beta_1 \geq 0$, $\beta_2 > 0$ *be arbitrary but fixed. Then the approximate solution* $\mathbf{u}_\varepsilon$ *produced by Algorithm 1 for* $\varepsilon < \varepsilon_0$ *satisfies*

$$|\mathrm{rank}(\mathbf{u}_\varepsilon)|_\infty \leq \big(d_{\mathbf{u}}^{-1} \ln\big[2(\alpha\kappa_1)^{-1}\rho_{\gamma_{\mathbf{u}}}\,\|\mathbf{u}\|_{\mathcal{A}_{\mathcal{H}}(\gamma_{\mathbf{u}})}\,\varepsilon^{-1}\big]\big)^{b_{\mathbf{u}}} \lesssim (|\ln\varepsilon| + \ln d)^{b_{\mathbf{u}}}, \tag{3.25}$$

$$\sum_{i=1}^{d} \#\,\mathrm{supp}_i(\mathbf{u}_\varepsilon) \lesssim d^{1+s^{-1}} \left(\sum_{i=1}^{d} \|\pi^{(i)}(\mathbf{u})\|_{\mathcal{A}^s}\right)^{\frac{1}{s}} \varepsilon^{-\frac{1}{s}}, \tag{3.26}$$

*as well as*

$$\|\mathbf{u}_\varepsilon\|_{\mathcal{A}_{\mathcal{H}}(\gamma_{\mathbf{u}})} \lesssim \sqrt{d}\,\|\mathbf{u}\|_{\mathcal{A}_{\mathcal{H}}(\gamma_{\mathbf{u}})}, \tag{3.27}$$

$$\sum_{i=1}^{d} \|\pi^{(i)}(\mathbf{u}_\varepsilon)\|_{\mathcal{A}^s} \lesssim d^{1+\max\{1,s\}} \sum_{i=1}^{d} \|\pi^{(i)}(\mathbf{u})\|_{\mathcal{A}^s}. \tag{3.28}$$

*The multiplicative constant in* (3.27) *depends only on* $\alpha$, *those in* (3.26) *and* (3.28) *depend only on* $\alpha$ *and* $s$.

*If in addition, Assumption 3.8 hold, then for the number of required operations* $\mathrm{ops}(\mathbf{u}_\varepsilon)$, *we have the estimate*

$$\mathrm{ops}(\mathbf{u}_\varepsilon) \leq C d^a\, d^{cs^{-1}\ln d} d^{12c\ln\ln d} |\ln\varepsilon|^{2c\ln d + 2\max\{b_{\mathbf{u}}, b_{\mathbf{f}}\}}\, \varepsilon^{-\frac{1}{s}}, \tag{3.29}$$

*where* $C, a$ *are constants independent of* $\varepsilon$ *and* $d$, *and* $c$ *is the smallest* $d$*-independent value such that* $I \leq c \ln d$ *for* $I$ *as in* (3.5). *In particular,* $c$ *does not depend on* $\varepsilon$ *and* $s$.

As in [3], the proof of Theorem 3.9 has two main constituents. On the one hand, one can use Theorem 3.3 in complete analogy to the use of Theorem 6.8 in [3]. On the other hand, one has to control $L(\mathbf{v})$ in (3.23). On account of (3.24), this can be done exactly as in ([3], Sects. 6.4, 6.5).

While the theoretical bounds have the same structure as for the scheme in [3], the concrete values of the constants are different and in fact more favorable (mainly due to the smaller exponents in (3.20) and (3.22)), as shown also by the numerical experiments discussed in the next section.

## 4. Numerical realization

### 4.1. Approximate application of operators

We now describe some practical improvements for the approximate application of operators in low-rank form required in Algorithm 1. Recall that for given compactly supported $\mathbf{v}$ and tolerance $\eta > 0$, we determine a suitable approximation $\tilde{\mathbf{T}}$ of $\mathbf{T}$ as well as an $n$ such that $\|\mathbf{P}\mathbf{T}\mathbf{v} - \mathbf{P}_n\tilde{\mathbf{T}}\mathbf{v}\| \le \eta$.

For our complexity estimates, we have assumed the choice of the parameter $n$ to be based directly on Theorem 2.5. This choice depends only on $\eta$ and on the maximum wavelet level in the support of $\mathbf{v}$, that is, on $\max_{\nu\in\operatorname{supp}\mathbf{v}}\max_i|\nu_i|$. We may, however, use the estimates in Theorem 2.5 in a slightly different way to take the actual values of $\mathbf{v}$ into account, and hence make use of additional *a posteriori* information.

According to (3.8) we first choose, independently of $n$, a suitable $\tilde{\mathbf{T}}$ such that $(1+\delta)\|\mathbf{S}^{-2}(\mathbf{T}-\tilde{\mathbf{T}})\mathbf{v}\| \le \frac{\eta}{2}$. It then remains to pick $n$ such that $\|(\mathbf{P}-\mathbf{P}_n)\tilde{\mathbf{T}}\mathbf{v}\| \le \frac{\eta}{2}$; here we can simply take into account the concrete values of $\tilde{\mathbf{T}}\mathbf{v}$ by noting that

$$\|(\mathbf{P}-\mathbf{P}_n)\tilde{\mathbf{T}}\mathbf{v}\| \le \max_\nu|p_\nu - p_{n,\nu}|\|\tilde{\mathbf{T}}\mathbf{v}\|.$$

In view of (2.12), it thus suffices to take

$$n = \left\lceil h^{-1}\left|\ln\left(\frac{\omega_{\min}^2\eta}{\|\tilde{\mathbf{T}}\mathbf{v}\|}\right)\right|\right\rceil.$$

This choice of $n$ is typically substantially smaller than the theoretical upper bounds in Theorem 3.3, where we needed to take additional measures to bound $\|\tilde{\mathbf{T}}\mathbf{v}\|$ and hence started instead from an estimate of the form $\|(\mathbf{P}-\mathbf{P}_n)\tilde{\mathbf{T}}\mathbf{v}\| \le \|(\mathbf{P}-\mathbf{P}_n)\mathbf{S}^2\,\mathrm{R}_{\operatorname{supp}\tilde{\mathbf{T}}\mathbf{v}}\|\|\mathbf{S}^{-2}\tilde{\mathbf{T}}\mathbf{v}\|$.

For the evaluation of $\mathbf{P}_n\tilde{\mathbf{T}}\mathbf{v}$, we additionally use a scheme analogous to the one described in ([3], Sect. 7.2) to add terms incrementally with additional tensor truncations, but preserving the total accuracy tolerance. To this end, we adjust the approximate operator evaluation such that $\mathbf{P}_n\tilde{\mathbf{T}} =: \mathbf{w}_{\eta/2}$ satisfies $\|\mathbf{P}\mathbf{T}\mathbf{v} - \mathbf{w}_{\eta/2}\| \le \eta/2$, and then determine an approximation $\tilde{\mathbf{w}}_{\eta/2}$ with $\|\mathbf{w}_{\eta/2} - \tilde{\mathbf{w}}_{\eta/2}\| \le \eta/2$, which is subsequently used as the output of $\textsc{apply}(\mathbf{v};\eta)$. With $\mathbf{P}_n = \sum_{\ell=1}^{\hat m(n)}\Theta_\ell$ and $\tilde{\mathbf{t}} := \tilde{\mathbf{T}}\mathbf{v}$, we first evaluate $\tau_\ell := \|\Theta_\ell\tilde{\mathbf{t}}\|$ for each $\ell$, build the ascendingly sorted sequence $\hat\tau_q := \tau_{\ell(q)}$, and find $q_0$ such that $\sum_{q=1}^{q_0}\hat\tau_q \le \eta/4$. The remaining contributions $\Theta_{\ell(q)}\tilde{\mathbf{t}}$ for $q = q_0+1,\dots,\hat m(n)$ are then summed in increasing order, with an application of $\textsc{recompress}(\cdot;\zeta_q)$ after adding each summand, with $\sum_{q=q_0+1}^{\hat m(n)}\zeta_q \le \eta/4$. At this point, we deviate slightly from the treatment in [3], and choose $\zeta_q$ using *a posteriori* information: as a by-product of $\textsc{recompress}(\cdot;\zeta_q)$, we obtain an estimate $\tilde\zeta_q$ of the actual truncation error, where usually $\tilde\zeta_q < \zeta_q$. To make use of this, we set $\tilde\eta_{q_0+1} := \eta/4$, and for each $q \ge q_0+1$ take $\zeta_q := \tilde\eta_q\hat\tau_q/\sum_{p=q}^{\hat m(n)}\hat\tau_p$ and $\tilde\eta_{q+1} := \eta_q - \tilde\zeta_q$. In this manner, truncation tolerances are again assigned in dependence on the relative sizes of summands.

### 4.2. Numerical experiments

In our numerical tests, we first treat the same high-dimensional Poisson problem as in [3] to allow a direct comparison to the algorithm with convergence enforced in $\mathrm{H}^1$-norm that we considered there. Subsequently, we apply the new scheme to a problem with tridiagonal diffusion matrix $M$. As in [3], we use $\mathrm{L}_2$-orthonormal, continuously differentiable, piecewise polynomial Donovan–Geronimo–Hardin multiwavelets [14] of polynomial

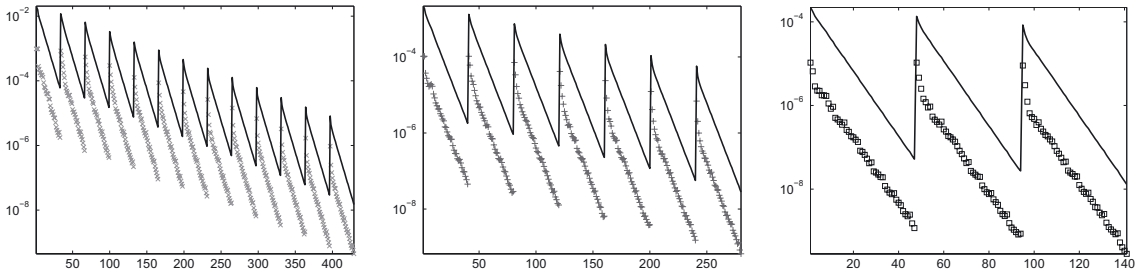


FIGURE 1. Norms of computed residual estimates (markers) and corresponding error bounds (lines), in dependence on the total number of inner iterations (horizontal axis), for $d = \times 16, +64, \square 256$.

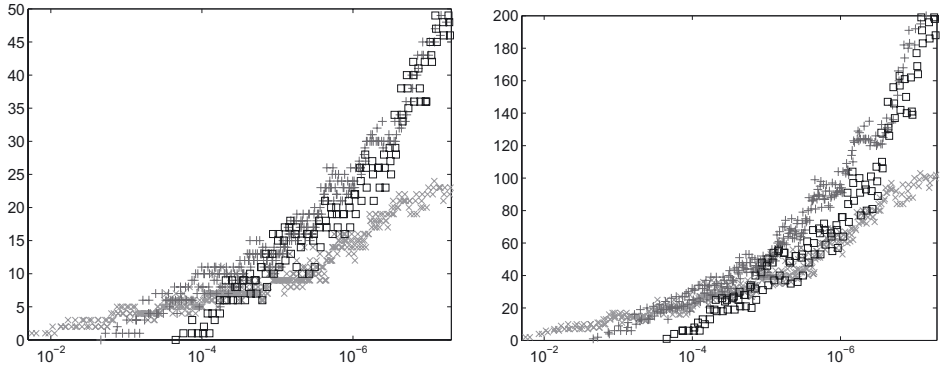


FIGURE 2. $|\mathrm{rank}(\mathbf{w}_{k,j})|_\infty$ (*left*) and maximum ranks of all intermediates arising in the inner iteration steps (*right*), in dependence on current estimate for $\|\mathbf{u} - \mathbf{w}_{k,j}\|$ (horizontal axis), for $d = \times 16, +64, \square 256$.

degree 6 and approximation order 7, which satisfy the conditions mentioned in Remark 2.2 and thus form a Riesz basis of $H^2(0,1) \cap H_0^1(0,1)$ after rescaling.

### *4.2.1. High-dimensional poisson problem*

Figures 1, 2, and 3 show the results for the Poisson problem on $(0,1)^d$ with homogeneous Dirichlet boundary conditions and right hand side $f = 1$. In comparison to the results obtained in [3], we generally observe a similar behavior, with the expected residual reduction and with ranks increasing gradually as the accuracy increases. One also observes periodic deteriorations of the error bounds due to the recompression and coarsening step in line 13 of Algorithm 1 which, however, guarantees overall a near-optimal balance of ranks and mode frame sparsity with the current error tolerance. The computational simplifications in the new scheme are apparent in Figure 3: with similar operation counts and error bounds, we can now go up to $d = 256$ instead of $d = 64$. However, the price to pay is that all error estimates now correspond to the $L_2$-norm, instead of the $H^1$-norm as in [3]. As illustrated in Figure 4, where we compare $L_2$- and $H^1$-errors to a reference solution computed by a highly accurate exponential sum approximation [15,18], we indeed no longer have control over the error in $H^1$ in the present case, but do obtain an upper bound for the $L_2$-error as guaranteed by our theory.

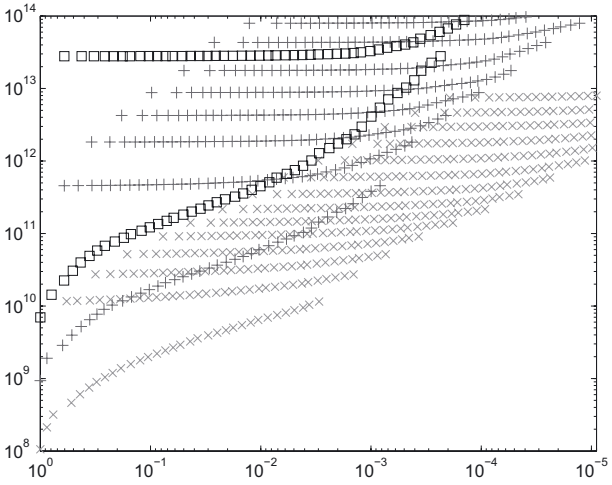


FIGURE 3. Operation count in dependence on the error estimate reduction (horizontal axis), for $d = \times 16, +64, \square 256$.

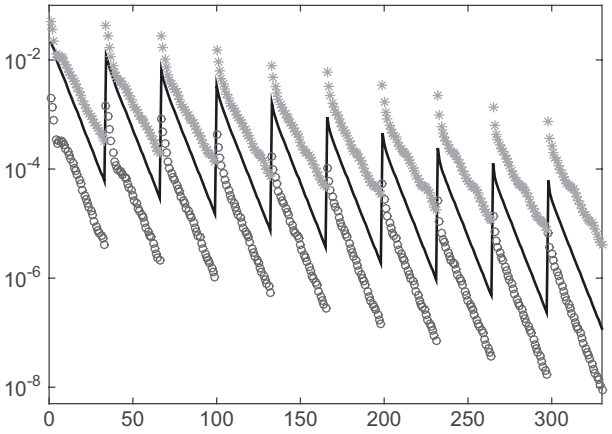


FIGURE 4. Computed error bound (lines), differences in $L_2$ ($\circ$) and $H^1$ ($*$) to reference solution, in dependence on the total number of inner iterations (horizontal axis), for $d = 16$.

*4.2.2. Dirichlet problem with tridiagonal diffusion matrix*

We now consider the case of tridiagonal diffusion matrices

$$M = (m_{ij})_{i,j=1,\dots,d} = \text{tridiag}(-a, 2, -a)$$

for $a = \frac{1}{2}$ and $a = 1$. As noted in ([3], Sect. 7.4), there is a significant difference in the behavior of the iteration and in the expected tensor approximability of the solution for these two values of $a$, since for $0 \le a < 1$, the condition number of $\mathbf{S}^{-1}\mathbf{T}\mathbf{S}^{-1}$ (which directly affects the lower bound for $\tilde{\rho}$ in the present scheme) remains bounded independently of $d$, whereas it grows proportionally to $d^2$ for $a = 1$. Figure 5 shows how this fact already manifests itself in a pronounced difference in the respective solution ranks observed for $d = 4$. The $d$-dependent condition number for $a = 1$ also leads to a substantial deterioration in the convergence of the iteration as $d$ increases. For $a = \frac{1}{2}$, however, we are still able to treat large values of $d$, as shown in Figures 6 and 7.

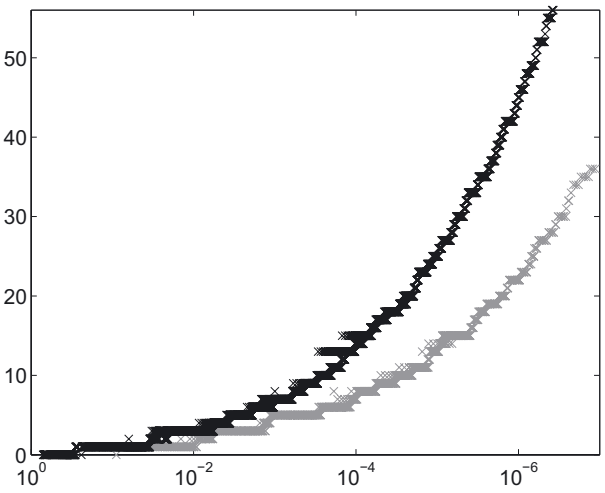


FIGURE 5. Tridiagonal diffusion matrix, $|\mathrm{rank}(\mathbf{w}_{k,j})|_\infty$ in dependence on current estimate for $\|\mathbf{u}-\mathbf{w}_{k,j}\|$ (horizontal axis), for $d=4$ and $a=\frac{1}{2}$ ($\times$) and $a=1$ ($\times$).

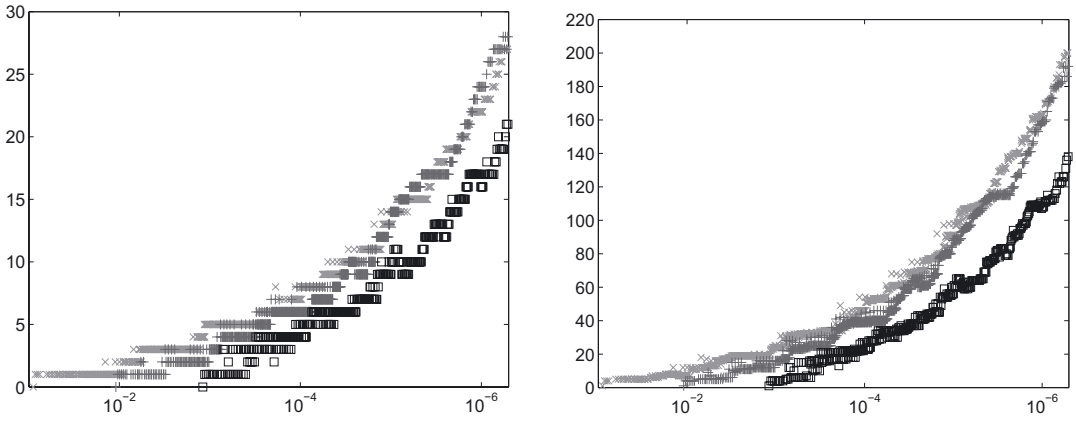


FIGURE 6. Tridiagonal diffusion matrix, $a=\frac{1}{2}$: $|\mathrm{rank}(\mathbf{w}_{k,j})|_\infty$ (*left*) and maximum ranks of all intermediates arising in the inner iteration steps (*right*), in dependence on current estimate for $\|\mathbf{u}-\mathbf{w}_{k,j}\|$ (horizontal axis), for $d=\times 4, +16, \square 64$.

In summary, we conclude that if convergence to the exact solution is required only in $L_2$, the seeming drawback of losing symmetry in the preconditioned system is more than compensated by the practical simplifications and by the gain in computational efficiency.

### 4.3. Comparison with a method for discretized problems

We conclude our numerical tests by a comparison to an established, conceptually different approach. For this comparison, we return to the Poisson problem $-\Delta u = 1$ with homogeneous Dirichlet boundary conditions on $(0,1)^d$ as in Section 4.2.1, where we again consider $d = 16, 64, 256$. As before, we denote by $u$ the exact solution of this partial differential equation.

Rather than adaptively adjusting the ranks in tandem with basis refinement as in Algorithm 1, all existing alternative methods that we are aware of operate on *fixed discretizations of the continuous problem* and,

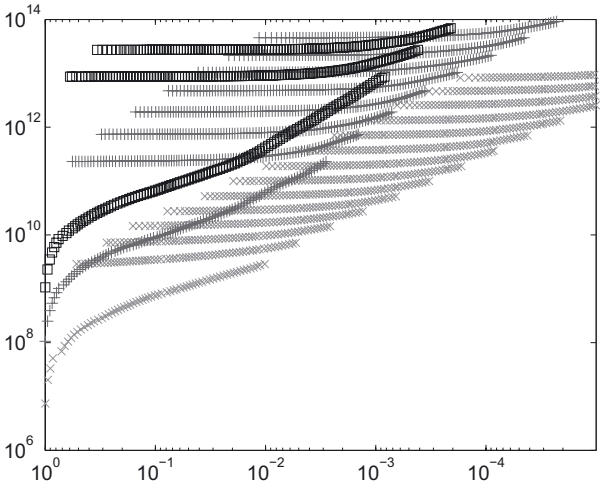


FIGURE 7. Tridiagonal diffusion matrix, $a = \frac{1}{2}$: operation count in dependence on the error estimate reduction (horizontal axis), for $d = \times 4, +16, \square 64$.

in contrast to our method, do not offer any mechanism for estimating the error with respect to $u$. These methods typically use finite difference discretizations on uniform grids, as for instance in [5, 13, 15, 24, 26]; in the case of the Poisson equation, for step size $h$ in each coordinate, one arrives at a discrete problem of the form $\mathbf{M}_h \mathbf{x}_h = \mathbf{f}_h$, where $\mathbf{f}_h = \mathbf{1}$ (denoting by $\mathbf{1}$ the vector with all entries equal to one) and where $\mathbf{M}_h = \mathbf{M}_h^{(1)} \otimes \mathbf{I} \otimes \ldots \otimes \mathbf{I} + \ldots + \mathbf{I} \otimes \ldots \otimes \mathbf{I} \otimes \mathbf{M}_h^{(1)}$, with $\mathbf{M}_h^{(1)} = h^{-2}\,\mathrm{tridiag}(-1, 2, -1)$ and $\mathbf{I}$ denoting the identity matrix of the same format.

Besides a comparison with conceptually more straightforward discretization schemes, our particular objective here is to establish a comparison with methods that make use of the particular structure of the underlying tensor representations. Prototypical examples are the *alternating linear scheme* (ALS), where single components in the tensor representation are optimized in an alternating fashion, and the related *density matrix renormalization group* (DMRG) algorithm. For an overview of such methods and further references, we refer to ([17], Sect. 3.2).

Specifically, we compare our method here with the *alternating minimal energy* (AMEn) method proposed recently by Dolgov and Savostyanov in [13], which is related to the ALS and DMRG methods. However, unlike the basic ALS scheme it includes a mechanism for automatically choosing appropriate ranks, and it was observed to perform better in the tests in [13] than the DMRG method. The AMEn scheme operates on *tensor train* (TT) representations [27, 28] that are closely related to the hierarchical Tucker representations used in our method.

### *4.3.1. A priori error estimates*

A direct comparison of methods makes sense only when the respective results are of comparable quality. In the present setting, the results thus need to meet comparable accuracy criteria with respect to the exact solution $u$ of the PDE. Unfortunately, assessing the output quality of any of the existing methods based on fixed discretizations of the continuous problem in terms of accuracy with respect to $u$ is severely hampered by the lack of rigorous *a posteriori* error estimates. Hence, one needs to resort to *a priori* estimates in order to obtain at least an indication of the *discretization error* that can be expected in dependence on $h$. In addition, one incurs an *interpolation error* by representing functions by grid values. Finally, these errors need to be balanced with the *solver error* in the approximate solution of the discretized system. Note that in Algorithm 1, there is no need to treat an interpolation error separately due to the use of an orthonormal basis, and the discretization and solver errors are balanced automatically and combined in the computed bounds on the total $L_2$-error with respect to $u$.

Returning to the above finite difference approximation, let $\mathbf{x}$ denote the array comprised of the grid values of the exact solution $u$. Then by Taylor expansion, one has the standard consistency error estimate

$$\|\mathbf{M}_h\mathbf{x} - \mathbf{f}_h\|_\infty \lesssim \sum_{i=1}^{d} \|\partial_{x_i}^4 u\|_\infty h^2 \leq d \max_i \|\partial_{x_i}^4 u\|_{\mathrm{L}_\infty(\Omega)} h^2.$$

To obtain an actual approximation in $\mathrm{L}_2(\Omega)$, the grid values $\mathbf{x}_h$ need to be extended to a function $u_h$. Here, in order to preserve the order of the above estimate, it is natural to employ piecewise $d$-linear interpolation. For a given $w \in \mathrm{H}^2(\Omega) \cap \mathrm{C}(\bar{\Omega})$, denoting this interpolant by $I_h w$, by ([8], Thm. 4.6.14) one has

$$\|w - I_h w\|_{\mathrm{L}_2(\Omega)} \leq C_d h^2 \left( \sum_{i=1}^{d} \left\| \partial_{x_i}^2 w \right\|_{\mathrm{L}_2(\Omega)}^2 \right)^{\frac{1}{2}}. \tag{4.1}$$

Although the dependence of $C_d$ on $d$ is not specified, considering simple examples of the form $w(x) = \sum_{i=1}^d f(x_i)$ for suitable $f \in \mathrm{C}^2([0,1])$, one finds that in general $C_d \gtrsim \sqrt{d}$.

Estimate (4.1) can be applied to $\|u - I_h u\|_{\mathrm{L}_2(\Omega)}$. Furthermore, since $\|\mathbf{M}_h^{-1}\|_2 \lesssim d^{-1}$, one has

$$\|I_h u - u_h\|_{\mathrm{L}_2(\Omega)} \lesssim h^{d/2} \|\mathbf{x} - \mathbf{x}_h\|_2 \leq \|\mathbf{M}_h^{-1}\|_2 h^{d/2} \|\mathbf{M}_h\mathbf{x} - \mathbf{f}_h\|_2 \lesssim d^{-1} \|\mathbf{M}_h\mathbf{x} - \mathbf{f}_h\|_\infty.$$

Consequently, we obtain

$$\|u - u_h\|_{\mathrm{L}_2(\Omega)} \lesssim \max_i \|\partial_{x_i}^4 u\|_{\mathrm{L}_\infty(\Omega)} h^2 + C_d \left( \sum_{i=1}^{d} \left\| \partial_{x_i}^2 w \right\|_{\mathrm{L}_2(\Omega)}^2 \right)^{\frac{1}{2}} h^2.$$

Since the constants in these estimates are not known, we do *not* obtain rigorous quantitative upper bounds on $\|u - u_h\|_{\mathrm{L}_2(\Omega)}$, but rather an indication of its approximate *order of magnitude*, assuming in particular that the involved quantities do not grow too strongly with $d$. Regarding the discretization error $\|I_h u - u_h\|_{\mathrm{L}_2(\Omega)}$, the latter assumption is in fact supported for the particular PDE under consideration by some numerical evidence, since in this specific case $h^{d/2}\|\mathbf{x} - \mathbf{x}_h\|_2$ can be estimated by comparison to a reference solution obtained by exponential sum approximation, similarly as for Figure 4. In this way one observes a scaling of order $\mathcal{O}(h^2)$ for this error component, with hardly any variation of the prefactor as $d$ increases. This means in particular that this scaling is also what one can expect *at best*, that is, the behaviour of the error does not improve with larger $d$ either.

For the interpolation error, the situation is less clear. If we assume that $\sum_{i=1}^d \left\|\partial_{x_i}^2 u\right\|_{\mathrm{L}_2(\Omega)}^2$ remains bounded independently of $d$, the considerations concerning estimate (4.1) suggest an interpolation error estimate of order $\mathcal{O}(\sqrt{d}h^2)$.

These error estimates now need to be balanced with the solver error, which can in turn be bounded in terms of the discrete residual $\|\mathbf{M}_h\tilde{\mathbf{x}}_h - \mathbf{f}_h\|_2$. Denoting again by $u_h$ the $d$-linear interpolant of the grid values $\mathbf{x}_h$ of the discrete solution, and by $\tilde{u}_h$ the interpolant for arbitrary given grid values $\tilde{\mathbf{x}}_h$, one has

$$\|u_h - \tilde{u}_h\|_{\mathrm{L}_2(\Omega)} \lesssim h^{d/2} \|\mathbf{x}_h - \tilde{\mathbf{x}}_h\|_2 \leq \|\mathbf{f}_h\|_2^{-1} \|\mathbf{x}_h - \tilde{\mathbf{x}}_h\|_2.$$

Since $\|\mathbf{x}_h - \tilde{\mathbf{x}}_h\|_2 \leq \|\mathbf{M}_h^{-1}\|_2 \|\mathbf{M}_h\tilde{\mathbf{x}}_h - \mathbf{f}_h\|_2$ with $\|\mathbf{M}_h^{-1}\|_2 \lesssim d^{-1}$, one obtains

$$\|u_h - \tilde{u}_h\|_{\mathrm{L}_2(\Omega)} \lesssim d^{-1} \frac{\|\mathbf{M}_h\tilde{\mathbf{x}}_h - \mathbf{f}_h\|_2}{\|\mathbf{f}_h\|_2}.$$

For the present model problem, the discretized problems thus should be solved up to a discrete residual of at most $d$ times the expected size of the further error contributions. This has been used as a stopping criterion for all following tests with such finite difference discretizations.

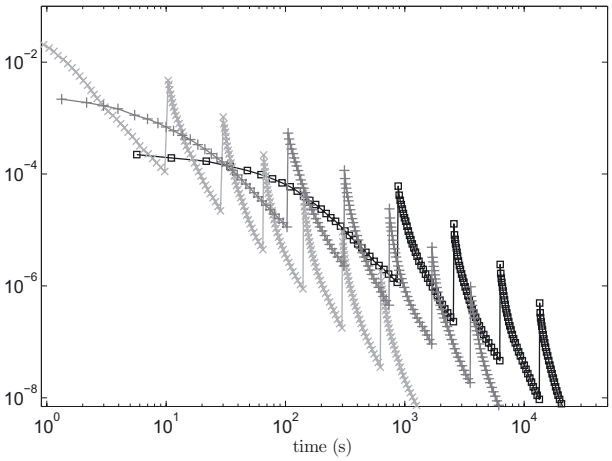


FIGURE 8. Error bound in dependence on wall clock time for Algorithm 1 as in Table 1, $d = \times 16, +64, \square 256$.

It needs to be stressed again that generally, *a priori* estimates as above for large $d$ do not by themselves give any reliable indication on the actual errors, since the involved constants depend both on $d$ and on the unknown solution. To arrive at suitable concrete residual tolerances that balance the involved errors, we will thus need to make some assumption on the actual scaling of discretization and interpolation errors with respect to both $h$ and $d$. Moreover, to see how the performance depends on given accuracy goals, based on the preceding discussion we consider both an "optimistic" and a "pessimistic" assumption in our tests. In contrast, the explicit error bounds computed by Algorithm 1 require only information on the data of the problem, that is, on $\mathbf{A}$ and $\mathbf{f}$.

*4.3.2. Results of the comparison*

We now choose the parameters in Algorithm 1 differently from the previous tests: we take $\kappa_{\mathrm{P}} = 1$, $\kappa_{\mathrm{C}} = 1$, $\alpha = 1$, $\beta_1 = \beta_2 = \frac{1}{10}$, and $\theta = \frac{1}{5}$. We emphasize that the algorithm then still gives explicit error bounds, but the values for $\kappa_{\mathrm{P}}$ and $\kappa_{\mathrm{C}}$ are smaller than required for our theoretical rank and complexity bounds in Theorem 3.9. The latter being worst-case bounds, however, these smaller values still lead to an observed better practical performance of the algorithm.

Since the compared methods are quite different in nature, we concentrate on the scaling of overall wall clock times in dependence on $d$ and on the target accuracy. The evolution of the guaranteed $\mathrm{L}_2$-error bound produced by Algorithm 1 (implemented in C++) is shown in Figure 8. This and all further computations were run on a single core of a workstation with Intel Core i5-3470S CPU at 2.9 GHz and 8 GB RAM.

For the AMEn method, we use the implementation `amen_solve2` (MATLAB with C and Fortran extensions) by Dolgov and Savostyanov as described in [13] from the TT Toolbox[3] by I. Oseledets *et al.*, with an additional check of the total relative residual $\|\mathbf{M}_h \cdot -\mathbf{f}_h\|_2/\|\mathbf{f}_h\|_2$ before stopping the iteration. This check is performed only when a simpler criterion on certain approximately projected residuals obtained as by-products, which is originally used by this solver, is met. This criterion on projected residuals, however, turns out *not* to be sufficient for guaranteeing a desired bound on the total residual. For the input parameters of the algorithm, we use the given default values for $\mathrm{AMEn}_{(\mathrm{ALS})}$. We have not found any prescription for adjusting these parameters for different grid sizes and values of $d$. Note that the AMEn solver has no mechanism for reutilizing results from coarser discretizations, although this may in principle be realizable. For each grid size, the computation is therefore restarted from initial values. Since random starting values are used by default, the obtained results are not exactly reproducible.

[3]available at https://github.com/oseledets/TT-Toolbox, used in the version of Jan 30, 2015.

TABLE 1. Run times (seconds) for arriving at given guaranteed error bounds $\varepsilon$ with Algorithm 1.

| $\varepsilon$ | $d = 16$ | $d = 64$ | $d = 256$ |
|---|---|---|---|
| 9.5e$-$7 | 140.0 | 603.6 | 1 790.9 |
| 2.4e$-$7 | 284.6 | 1 332.6 | 2 554.6 |
| 6.0e$-$8 | 575.1 | 2 799.0 | 5 921.5 |
| 1.5e$-$8 | 1 073.7 | 5 533.3 | 12 358.6 |

A direct application of AMEn without preconditioning gives good results for coarse discretizations as reported in [13], but leads to very slow convergence for finer discretizations. These issues with ill-conditioned problems are also well-known for the related ALS and DMRG methods. Note that one cannot rely directly on preconditioners for the original full discrete problem in order to precondition the arising projected systems. This point is considered in [26] for eigenvalue problems, but we are not aware of any publication where this is integrated into a complete method for linear systems. Regardless of this preconditioning issue, for higher accuracy requirements such a direct approach based low-order finite differences becomes expensive in comparison with our high-order wavelet discretization, simply as a consequence of the number of entailed degrees of freedom in each tensor mode.

Both problems can be alleviated to some degree by a *QTT representation* [23] where, assuming a tensor product discretization with $2^L$ grid points in each coordinate, the corresponding tensors of order $d$ are reinterpreted as tensors of order $dL$ with mode sizes 2 and, in this form, decomposed in a tensor train representation. This approach is also used for certain examples in [13]. Provided that the ranks in the resulting representation remain small, one can obtain a significant further compression. The ill-conditioning of $\mathbf{M}_h$ also appears to present less of a problem in this case, at least as long as the ranks do not become too large.

Therefore, we use the AMEn solver combined with a QTT representation of the problem for our comparison. The underlying discretization uses standard finite differences with step size $h$ as described above, where $h = (2^L + 1)^{-1}$ for some $L \in \mathbb{N}$; recall that in this case, tensor entries represent point values of the approximated functions.

Note that in the subsequent experiments, the absolute wall clock times need to be interpreted with caution, in particular since the test code of Algorithm 1 has not been optimized for execution speed. We can, however, make some observations concerning the scaling of the total complexities both with respect to the target errors and with respect to $d$.

The results for Algorithm 1 are given in Table 1, and the results for AMEn/QTT are summarized in Tables 2 and 3. As mentioned earlier, using AMEn directly *without* the QTT representation and without a preconditioner would be substantially less efficient. For instance, for $d = 16$ and $L = 10$, a direct solution in the $d$th order TT representation of the tensor already takes 5138.0 s to converge to the same accuracy as in Table 3, with a very rapid increase of run times for larger $L$.

Since in the case of the AMEn scheme, we do not know the precise constants in the error estimates and their dependence on $d$, we now need to make an ad hoc assumption on the involved errors to arrive at suitable residual tolerances for the solver. On the one hand, we can make the optimistic assumption of an $\mathcal{O}(h^2)$ behaviour for the sum of discretization and interpolation error, without further $d$-dependence. By our previous considerations, this represents the best scaling with respect to $d$ that one could expect here. Although one cannot rule out that the error actually shows this behavior, it cannot be inferred from the *a priori* estimates either. This choice leads us to using $dh^2$ as the residual bound, with the results given in Table 2. On the other hand, the theoretically supported bounds derived in our above discussion suggest that it is reasonable to rather expect an error estimate of order $\mathcal{O}(\sqrt{d}\,h^2)$. Balancing the error components then requires solving up to a discrete residual of $d\sqrt{d}\,h^2$. The results for this case are displayed in Table 3.

Especially under the first assumption, in the case of the AMEn solver, the computational costs are influenced by the spatial dimension $d$ only rather weakly. This is a consequence of the special structure of the problem,

TABLE 2. Run times for AMEn/QTT at discretization levels $L$, assuming an error estimate $\mathcal{O}(h^2)$ (solving up to a discrete residual $dh^2$).

| $L$ | $h^2$ | $d=16$ | $d=64$ | $d=256$ |
|---|---|---|---|---|
| 10 | 9.5e$-$7 | 31.6 | 88.6 | 83.4 |
| 11 | 2.4e$-$7 | 95.9 | 226.7 | 268.5 |
| 12 | 6.0e$-$8 | 789.9 | 814.5 | 798.4 |
| 13 | 1.5e$-$8 | 6 315.0 | 4 304.8 | 2 380.3 |

TABLE 3. Run times for AMEn/QTT at discretization levels $L$, assuming an error estimate $\mathcal{O}(\sqrt{d}\,h^2)$ (solving up to a discrete residual $d\sqrt{d}\,h^2$).

| $d=16$ | | | $d=64$ | | | $d=256$ | | |
|---|---|---|---|---|---|---|---|---|
| $L$ | $\sqrt{d}\,h^2$ | | $L$ | $\sqrt{d}\,h^2$ | | $L$ | $\sqrt{d}\,h^2$ | |
| 10 | 9.5e$-$7 | 71.4 | 11 | 1.9e$-$6 | 134.2 | 12 | 9.5e$-$7 | 321.9 |
| 11 | 2.4e$-$7 | 540.2 | 12 | 4.8e$-$7 | 463.1 | 13 | 2.4e$-$7 | 852.5 |
| 12 | 6.0e$-$8 | 3 783.5 | 13 | 1.2e$-$7 | 2 778.5 | 14 | 6.0e$-$8 | 4 037.0 |
| 13 | 1.5e$-$8 | 27 746.3 | 14 | 3.0e$-$8 | 21 599.2 | 15 | 1.5e$-$8 | —* |
| | | | 15 | 7.5e$-$9 | 154 539.1 | | | |

*Out of memory in global residual evaluation.

because $\|u\|_{L_2} \lesssim d^{-1}$ and the required residual bounds are weakened when $d$ increases. The scheme is clearly able to effectively exploit these properties.

Let us now turn to the scaling with respect to the expected error in the case of AMEn. Passing from $L$ to $L+1$ amounts to replacing $h$ by $h/2$, which we can expect to reduce the total error by a factor 4. The observed corresponding increases in the wall clock times between $L$ and $L+1$ are fairly irregular, with the largest relative changes occurring at larger values of $L$. The costs can also be seen to depend quite strongly on the assumptions made on discretization and interpolation errors, which underscores the need for a careful balancing of different error contributions. The results also hint at ill-conditioning issues becoming more problematic, even for the QTT representation, when $L$ and the required approximation ranks are large.

In the case of Algorithm 1, the results with guaranteed error bounds $\varepsilon$ are given in Table 1. We observe a scaling of the costs with respect to $d$ that is at first slightly superlinear and subsequently sublinear for larger $d$. Our method can thus profit from the particular features of the problem for large $d$ as well, albeit not quite as strongly as the AMEn scheme.

Regarding the desired target accuracy, Algorithm 1 shows a scaling that is very regular and substantially better than in the case of the AMEn solver. For instance, for $d=16$, the repeated decrease of the error tolerance by these factors of 4 leads to an increase of the computational time by factors 2.03, 2.02, 1.87; one finds a similar behaviour for larger $d$. The convergence thus accelerates slightly as the error decreases. This is to be expected in view of the bound (3.29): asymptotically, the factor corresponding to the total number of activated basis elements in the tensor modes, which in the present example is expected to grow as $\varepsilon^{-\frac{1}{7}}$, will dominate the further logarithmic terms in this complexity estimate. Indeed, as shown in Figure 9, we obtain this expected asymptotic behavior for the basis size, which becomes evident for $\varepsilon < 10^{-5}$. The largest ranks of tensors arising in the computation can be seen to grow like a multiple of $|\log\varepsilon|^2$. Clearly, the overhead caused by adaptivity is more visible for smaller problem sizes.

A further difference of practical importance between the two methods is that the use of point values in the finite difference discretization can lead to overflows in high dimensions. In the present example this can be observed for $\mathbf{f}_h = \mathbf{1}$, for which $\|\mathbf{f}_h\|_2$ becomes larger than the greatest representable value in the tests with $d=256$. This issue can be circumvented by a careful rescaling of tensor components. Note, however, that as

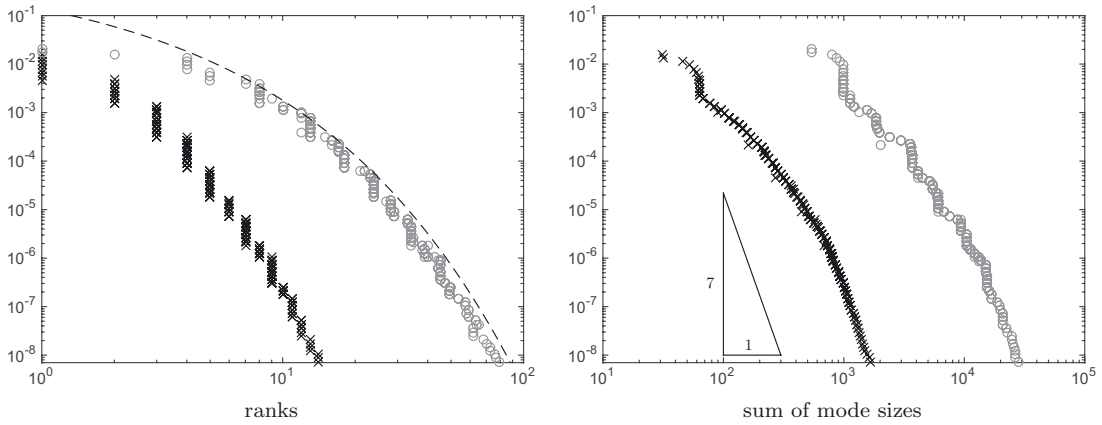


FIGURE 9. Error bound in dependence of maximum hierarchical ranks and basis sizes for Algorithm 1 as in Figure 8, $d = 16$. Here $\times$ represent quantities for the solutions $\mathbf{w}_{k,j}$ and $\circ$ the largest respective quantities arising in intermediate residual evaluation steps. *Left:* maximum hierarchical ranks, where the dashed line is $\exp(-2\sqrt{\cdot})$; *right:* total number of active basis elements, corresponding to the sum of mode sizes of the tensors.

long as the $L_2$-norms of the represented functions do not become very large with increasing $d$, this problem cannot arise when using orthonormal bases as in Algorithm 1.

In summary, for lower accuracy requirements and correspondingly smaller discretized problems, the costs for both methods remain comparable, although the overhead for adaptivity and error estimation carries a certain weight in this regime. In particular, for more demanding accuracy requirements, however, our method eventually becomes clearly superior.

*Acknowledgements.* The authors would like to thank Kolja Brix for providing multiwavelet construction data used in the numerical experiments.

## REFERENCES

[1] R. Andreev and C. Tobler, Multilevel preconditioning and low rank tensor iteration for space-time simultaneous discretizations of parabolic PDEs. *Numer. Linear Algebra Appl.* **22** (2015) 317–337.

[2] M. Bachmayr, *Adaptive Low-Rank Wavelet Methods and Applications to Two-Electron Schrödinger Equations*. Ph.D. thesis, RWTH Aachen (2012).

[3] M. Bachmayr and W. Dahmen, Adaptive low-rank methods: Problems on Sobolev spaces. Preprint arXiv:1407.4919 [math.NA] (2014).

[4] M. Bachmayr and W. Dahmen, Adaptive near-optimal rank tensor approximation for high-dimensional operator equations. *Found. Comput. Math.* **15** (2015) 839–898.

[5] J. Ballani and L. Grasedyck, A projection method to solve linear systems in tensor format. *Numer. Linear Algebra Appl.* **20** (2013) 27–43.

[6] G. Beylkin and L. Monzón, Approximation by exponential sums revisited. *Appl. Comput. Harmon. Anal.* **28** (2010) 131–149.

[7] M. Billaud-Friess, A. Nouy and O. Zahm, A tensor approximation method based on ideal minimal residual formulations for the solution of high-dimensional problems. *ESAIM: M2AN* **48** (2014) 1777–1806.

[8] S.C. Brenner and L.R. Scott, The Mathematical Theory of Finite Element Methods. 3rd edition. Vol. 15 of *Texts Appl. Math.* Springer (2008)

[9] A. Cohen, W. Dahmen and R. DeVore, Adaptive wavelet methods for elliptic operator equations: Convergence rates. *Math. Comput.* **70** (2001) 27–75.

[10] W. Dahmen, Stability of multiscale transformations. *J. Fourier Anal. Appl.* **2** (1996) 341–361.

[11] W. Dahmen, R. DeVore, L. Grasedyck and E. Süli, Tensor-sparsity of solutions to high-dimensional elliptic partial differential equations. *Found. Comput. Math.* (2015) DOI: 10.1007/s10208-015-9265-9.

[12] T.J. Dijkema, C. Schwab and R. Stevenson, An adaptive wavelet method for solving high-dimensional elliptic PDEs. *Constr. Approx.* **30** (2009) 423–455.

[13] S.V. Dolgov and D.V. Savostyanov, Alternating minimal energy methods for linear systems in higher dimensions. *SIAM J. Sci. Comput.* **36** (2014) A2248–A2271.

[14] G.C. Donovan, J.S. Geronimo and D.P. Hardin, Orthogonal polynomials and the construction of piecewise polynomial smooth wavelets. *SIAM J. Math. Anal.* **30** (1999) 1029–1056.

[15] L. Grasedyck, Existence and computation of low Kronecker-rank approximations for large linear systems of tensor product structure. *Computing* **72** (2004) 247–265.

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

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

[18] W. Hackbusch, *Entwicklungen nach Exponentialsummen*. Technical Report 4, MPI Leipzig (2005).

[19] W. Hackbusch, Tensor Spaces and Numerical Tensor Calculus. Vol. 42 of *Springer Series Comput. Math.* Springer-Verlag Berlin Heidelberg (2012).

[20] W. Hackbusch and B.N. Khoromskij, Low-rank Kronecker-product approximation to multi-dimensional nonlocal operators. Part I. Separable approximation of multi-variate functions. *Computing* **76** (2006) 177–202.

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

[22] B.N. Khoromskij, Tensor-structured preconditioners and approximate inverse of elliptic operators in $\mathbb{R}^d$. *Constr. Approx.* **30** (2009) 599–620.

[23] B.N. Khoromskij, $O(d \log N)$-quantics approximation of $N$-d tensors in high-dimensional numerical modeling. *Constr. Approx.* **34** (2011) 257–280.

[24] D. Kressner and C. Tobler. Preconditioned low-rank methods for high-dimensional elliptic PDE eigenvalue problems. *Comput. Methods Appl. Math.* **11** (2011) 363–381.

[25] D. Kressner and A. Uschmajew, On low-rank approximability of solutions to high-dimensional operator equations and eigenvalue problems. Preprint arXiv:1406.7026 [math.NA] (2014).

[26] D. Kressner, M. Steinlechner and A. Uschmajew, Low-rank tensor methods with subspace correction for symmetric eigenvalue problems. *SIAM J. Sci. Comput.* **36** (2014) A2346–A2368.

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

[28] I. Oseledets and E. Tyrtyshnikov, Breaking the curse of dimensionality, or how to use SVD in many dimensions. *SIAM J. Scientific Comput.* **31** (2009) 3744–3759.

[29] F. Stenger, Numerical Methods Based on Sinc and Analytic Functions. Vol. 20 of *Springer Series Comput. Math.* Springer-Verlag (1993).

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