NUMERICAL LINEAR ALGEBRA WITH APPLICATIONS
Numer. Linear Algebra Appl. 2013; **00**:1–21
Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/nla

Multilevel preconditioning and low rank tensor iteration for space-time simultaneous discretizations of parabolic PDEs

Roman Andreev*† and Christine Tobler‡

SUMMARY

This paper addresses the solution of parabolic evolution equations simultaneously in space and time as may be of interest in e.g. optimal control problems constrained by such equations. As a model problem we consider the heat equation posed on the unit cube in Euclidean space of moderately high dimension. An a priori stable minimal residual Petrov-Galerkin variational formulation of the heat equation in space-time results in a generalized least squares problem. This formulation admits a unique, quasi-optimal solution in the natural space-time Hilbert space and serves as a basis for the development of space-time compressive solution algorithms.

The solution of the heat equation is obtained by applying the conjugate gradient method to the normal equations of the generalized least squares problem. Starting from stable subspace splittings in space and in time, multilevel space-time preconditioners for the normal equations are derived.

In order to reduce the complexity of the full space-time problem, all computations are performed in a compressed or sparse format called the hierarchical Tucker format supposing that the input data is available in this format. In order to maintain sparsity, compression of the iterates within the hierarchical Tucker format is performed in each conjugate gradient iteration. Its application to vectors in the hierarchical Tucker format is detailed.

Finally, numerical results in up to five spatial dimensions based on the recently developed `htucker` toolbox for MATLAB are presented.

Received . . .

1. INTRODUCTION

Parabolic evolution equations arise in many applications ranging from engineering to the social sciences. The standard and versatile numerical approach to solving such equations is the *method of lines*, or *time stepping*, which either reduces the problem to a system of coupled ordinary differential equations by means of a semidiscretization in space, or to a set of elliptic problems to be solved sequentially by means of a semidiscretization in time, see e.g. [1]. However, two fundamental issues, a practical and a theoretical one, motivate a *simultaneous space-time* discretization and solution of parabolic equations.

The practical issue is the parallelization of the solution process. An intrinsic limitation to the exploitation of growing parallel computer architectures in time stepping methods is the sequential dependence of the computed solution at intermediate time points on the previous ones.

†RICAM of the Austrian Academy of Sciences, 4040 Linz, Austria, roman.andreev@oeaw.ac.at. Research supported by the Swiss National Science Foundation grant No. PDFMP2-127034/1.
‡Mathworks, 3 Apple Hill Dr, Natick (MA) 01760, United States, christine.tobler@mathworks.com. Research supported by the Swiss National Science Foundation grant No. PDFMP2-124898.
*Correspondence to: roman.andreev@oeaw.ac.at

Several methods have been devised to cope with this limitation, see [2], for example. Still, some applications, such as optimal control problems with parabolic PDE constraints, may require the knowledge of the solution to a parabolic problem over the whole time horizon. Since the storage of the full solution in space and time may quickly become prohibitive for problems posed in several dimensions, some form of adaptivity that can be performed in parallel and simultaneously in space and time becomes essential. We refer to such algorithms as *space-time compressive* algorithms.

The theoretical issue is the optimality (in terms of minimal work for given target accuracy, possibly up to a multiplicative constant) of the algorithm. While convergence can be routinely established for time-stepping methods (and the derived methods mentioned in [2]), optimality seems to be exclusive to space-time methods. The reason may be sought in the hidden *elliptic* flavor of the parabolic equation which is revealed once it is formulated as a well-posed operator equation in suitable Bochner spaces, see [3, Cha. 3, Sects. 4-7]. Recent numerical methods which capitalize on this fact are the adaptive wavelet method [4, 5] and the a priori stable (nonadaptive) minimal residual Petrov-Galerkin discretization [6, 7, 8].

The adaptive wavelet method of [4] adopts the abstract operator equation perspective and, after a choice of suitable Riesz bases, reformulates the parabolic equation as an equivalent matrix-vector equation, where the vectors are elements of $\ell^2(\mathbb{N})$ and the parabolic operator is represented by a bi-infinite matrix. If the matrix can be suitably approximated by matrices with only finitely many nonzero entries – this is the case for many relevant parabolic equations – then the adaptive wavelet methods of [9, 10] yield optimal rates of convergence.

While the optimality of the adaptive wavelet method is understood, several of its ingredients are difficult to obtain in practice. First, the overall approach is fundamentally different to the finite element method, and therefore has to be programmed essentially from scratch. Second, and this is particularly true for parabolic problems, several requirements are imposed on the wavelet basis [4, Section 8.1] which make their construction difficult. Indeed, few numerical experiments have been published, with spatial dimension not more than two and with constant coefficients [5, 11].

The purpose of this paper is therefore to present and to discuss a practical space-time compressive discretization algorithm for the particular example of the heat equation on the unit cube in several spatial dimensions. The implementation relies on fairly standard components of the finite element method. While parallels to [4] can be identified, the first important and distinctive feature of the present algorithm is the fact that a pair of fixed finite-dimensional space-time trial and test spaces of tensor product type is used for the discretization. Unlike in the adaptive wavelet method, these are determined a priori and are shown to be stable, i.e., the parabolic operator satisfies an inf-sup condition on those spaces uniformly in the discretization level. A second ingredient is a pair of numerically accessible operators on the test and trial spaces which generate the "correct" norms. From these, a well-posed finite system of least squares type is obtained. The solution satisfies a quasi-optimality estimate in the natural spaces for the continuous equation (in no way dependent on the mesh), as in the Céa's lemma. This is the second important feature of the present algorithm. The norm generating operators proposed and tested here are derived from the well-known BPX preconditioner [12, 13, 14], several copies of which are combined into one "parabolic BPX preconditioner" acting in space-time. In this way, the intricate construction of suitable wavelets required in [4] is bypassed. This contributes to the practicality and the novelty of the algorithm.

Space-time compressive algorithms are crucial in applications with solutions featuring singularities and/or posed in high dimension. Here, we address the latter scenario. As already indicated, we compute in spaces of tensor product type. These are constructed by tensorization of univariate finite-dimensional spaces and the dimension of the trial and test spaces increases rapidly as we only consider uniform mesh refinement. However, at no point in the computation do we require the storage or the computation of the full vector of this size, which would be infeasible. This presupposes that the input data and the solution may be well-approximated in a *low rank tensor format*. For the solution, such an approximation is performed adaptively during the iterative solution process. Such formats include the "classical" CP and Tucker formats, see [15] for a review. We will work with the more recently developed *hierarchical Tucker format* [16, 17], since it admits

DOI: 10.1002/nla

efficient basic operations such as addition, inner product and multiplication by a matrix, as well as approximation of a low rank tensor by a lower-rank tensor.

Methods for the solution of linear systems, in particular stemming from evolution equations, in the hierarchical Tucker format, as well as in the Tensor Train format [18], have been the subject of recent work.

The first main approach [19, 20, 21, 22] is via developing low rank variants of iterative methods for the solution of linear systems, replacing the iterates by low rank tensors and using the basic operations described above. As the ranks of the iterates tend to grow dramatically during such an iteration, a recurrent approximation of the iterates by tensors of lower rank is employed. A good preconditioner is usually critical to the performance. The second approach [23, 24, 25, 26, 27] are the DMRG or (M)ALS methods, which work directly within the structure of the low rank tensor format, typically the hierarchical Tucker or the Tensor Train format. These methods are based on repeatedly projecting the high-dimensional linear system onto subspaces of lower dimension, and solving the resulting smaller systems. We follow the first route: the proposed BPX preconditioner renders the linear system well conditioned, while the system matrix does not readily lend itself to the projections necessary in the DMRG or (M)ALS methods. In [28, 29, 30, 31], other low rank tensor approaches for the solution of evolution equations are proposed. These are based on time-stepping or on the explicit representation of the solution using the exponential of the generator.

The outline of the paper is as follows. Section 2 introduces the model heat equation and the space-time variational formulation in certain Bochner spaces $\mathcal{X}$ and $\mathcal{Y}$. Then, a so-called minimal residual Petrov-Galerkin formulation is introduced which allows a stable space-time discretization of the heat equation in finite dimensional spaces. It relies on the availability of certain operators which generate equivalent norms on $\mathcal{X}$ and $\mathcal{Y}$. Following the methodology of operator preconditioning [32], these operators provide preconditioners for the resulting linear system, which has the form of generalized least squares equations, i.e., least squares with minimization w.r.t. specific norms. An example of such operators based on multilevel norm equivalences induced by orthogonal subspace splittings [33] is then constructed. Finite dimensional tensor product space-time test and trial spaces which conform with the minimal residual Petrov-Galerkin formulation are defined. Section 3 introduces the hierarchical Tucker format [17, 16], and describes the Kronecker product structure of the system matrix B resulting from the minimal residual Petrov-Galerkin discretization. The matrix B can be efficiently applied to a vector $\mathbf{x}$ stored as a low rank tensor in the hierarchical Tucker format. Approximate preconditioners are derived that can be applied to such a tensor efficiently. This allows the formulation of a variant of the preconditioned conjugate gradient method for the generalized least squares equations based on the hierarchical Tucker format. Section 4 discusses our numerical examples which comprise isotropic and anisotropic diffusion in the cube $(-1,1)^d$ of dimension $d = 1, \dots, 5$. See Section 5 for conclusions and outlook.

Let us comment on the notation used throughout. The symbol $\otimes$ denotes the tensor product of Hilbert spaces [34], as well as the Kronecker product of matrices. A disjoint union of two sets t and s is denoted by $t \dot\cup s$. An element $\mathbf{x} \in \mathbb{R}^{n_0 \times \cdots \times n_d}$ is called a tensor. We identify $\mathbf{x}$ with the vector of length $n_0 \cdots n_d$ where convenient. The norm $\|\mathbf{x}\|$ of a tensor $\mathbf{x}$ is defined as the Euclidean norm of the corresponding vector. As a rule, vectors are denoted by lowercase letters and matrices by uppercase letters. For a symmetric positive definite (s.p.d.) matrix M we use the notation $\mathsf{M}^{1/2}$ to denote a matrix such that $\mathsf{M} = \mathsf{M}^{\top/2}\mathsf{M}^{1/2}$, e.g. the Cholesky factor, where $(\cdot)^\top$ denotes matrix transposition.

2. PARABOLIC PDES

2.1. The model problem

In this section we introduce our model parabolic evolution equation which is the instationary diffusion in a cube. Let $0 < T_{\text{final}} < \infty$ and set $J := (0, T_{\text{final}})$. Let $D \subset \mathbb{R}^d$, where $d \in \mathbb{N}$ is fixed, be an open "spatial" domain with a Lipschitz boundary. In Section 2.4.2 we will specialize to the

DOI: 10.1002/nla

case $D = (-1,1)^d$. We consider the evolution equation

$$\partial_t x(t,\xi) - \operatorname{div}(q(t,\xi)\operatorname{grad} x(t,\xi)) = p(t,\xi), \qquad (t,\xi) \in J \times D, \tag{2.1}$$
$$x(0,\xi) = h(\xi), \qquad \xi \in D, \tag{2.2}$$
$$x(t,\xi) = 0, \qquad (t,\xi) \in J \times \partial D, \tag{2.3}$$

where $q \in L^\infty(J \times D)$ is a space- and time-dependent coefficient and the differential operators "div" and "grad" are w.r.t. the spatial variable $\xi \in D$. We assume

$$0 < a_{\min} := \operatorname*{ess\,inf}_{J\times D} q \le \operatorname*{ess\,sup}_{J\times D} q =: a_{\max} < \infty. \tag{2.4}$$

We introduce the standard Sobolev spaces $V := H_0^1(D)$ and $H := L^2(D)$. The norm on V is given by the $H^1(D)$ seminorm. The embedding $V \hookrightarrow H$ is continuous and dense. We denote the scalar product on H by $\langle\cdot,\cdot\rangle_H$ and the duality pairing on $V' \times V$ by $\langle\cdot,\cdot\rangle_{V'\times V}$, and similarly for other Hilbert spaces. We identify H with its dual H' via the scalar product $\langle\cdot,\cdot\rangle_H$. Thus, one can think of $\langle\cdot,\cdot\rangle_{V'\times V}$ as being the unique continuous extension of $\langle\cdot,\cdot\rangle_H : V \times V \to \mathbb{R}$. For (a.e.) $t \in J$ we define $a(t;\cdot,\cdot) : V \times V \to \mathbb{R}$ by

$$a(t;\nu,\bar\nu) = \int_D q(t,\xi)\operatorname{grad}\nu(\xi)\cdot\operatorname{grad}\bar\nu(\xi)d\xi \quad \text{for all} \quad \nu,\bar\nu \in V. \tag{2.5}$$

By (2.4), the symmetric bilinear form $a(t;\cdot,\cdot)$ is continuous and coercive with constants uniform in (a.e.) $t \in J$. Further, it follows from [35, Lemma 4.4.1] that for all $\nu,\bar\nu \in V$, the mapping $J \ni t \mapsto a(t;\nu,\bar\nu)$ is measurable.

In order to obtain a well-posed space-time variational formulation of (2.1)–(2.3) we follow [4] and introduce the Bochner spaces

$$\mathcal{X} := L^2(J,V) \cap H^1(J,V') \cong [L^2(J)\otimes V] \cap [H^1(J)\otimes V'],$$
$$\mathcal{Y} := \mathcal{Y}_1 \times \mathcal{Y}_2 := L^2(J,V) \times H \cong [L^2(J)\otimes V] \times H,$$

with norms $\|\cdot\|_{\mathcal{X}}$ and $\|\cdot\|_{\mathcal{Y}}$ given by

$$\|x\|_{\mathcal{X}}^2 := \|x\|_{L^2(J,V)}^2 + \|\partial_t x\|_{L^2(J,V')}^2, \quad \|y\|_{\mathcal{Y}}^2 := \|y_1\|_{L^2(J,V)}^2 + \|y_2\|_H^2, \tag{2.6}$$

for all $x \in \mathcal{X}$ and $y = (y_1,y_2) \in \mathcal{Y}$. One can view $x \in \mathcal{X}$ as either (an equivalence class of functions) $x : J \times D \to \mathbb{R}$ or as $x : J \to V'$, and we will frequently switch between these two interpretations. Let $\overline{J}$ denote the closure of J. We note that

$$\sup_{t\in\overline{J}} \sup_{x\in\mathcal{X}\setminus\{0\}} \frac{\|x(t)\|_H}{\|x\|_{\mathcal{X}}} < \infty, \tag{2.7}$$

i.e., $\mathcal{X} \hookrightarrow C^0(\overline{J},H)$ continuously, and in particular, the trace map $(\cdot)|_{t=0} : \mathcal{X} \to H$, $x \mapsto x|_{t=0} = x(0)$, is well-defined and continuous [3, Chapter 1]. We define the linear operator $\mathcal{B} : \mathcal{X} \to \mathcal{Y}'$ by

$$(\mathcal{B}x)(y) = \int_J \{\langle\partial_t x(t), y_1(t)\rangle_{V'\times V} + a(t;x(t),y_1(t))\}\,dt + \langle x(0), y_2\rangle_H \tag{2.8}$$

for all $x \in \mathcal{X}$ and $y = (y_1,y_2) \in \mathcal{Y}$, as well as the load functional $b : \mathcal{Y} \to \mathbb{R}$ by

$$b(y) = \int_J \langle p(t), y_1(t)\rangle_{V'\times V} dt + \langle h, y_2\rangle_H \tag{2.9}$$

for all $y = (y_1,y_2) \in \mathcal{Y}$. It is easy to check that $\mathcal{B} \in \mathcal{L}(\mathcal{X},\mathcal{Y}')$ and $b \in \mathcal{Y}'$. The space-time variational formulation of (2.1)–(2.3) now reads:

$$\text{find} \quad x \in \mathcal{X} \quad \text{s.t.} \quad (\mathcal{B}x)(y) = b(y) \quad \forall y \in \mathcal{Y}. \tag{2.10}$$

This variational formulation is well-posed by [4, Theorem 5.1], which we recall here for future reference.

Theorem 2.1. *The operator $\mathcal{B} \in \mathcal{L}(\mathcal{X},\mathcal{Y}')$ is an isomorphism.*

DOI: 10.1002/nla

2.2. *Minimal residual Petrov-Galerkin discretization*

In this section we derive a discrete space-time variational formulation of (2.1)–(2.3) from (2.10). It is based on operators $\mathcal{M}$ and $\mathcal{N}$ which generate norms on $\mathcal{X}$ and $\mathcal{Y}$, resp., and provide preconditioners for the discrete system.

Let $\mathcal{M} \in \mathcal{L}(\mathcal{X}, \mathcal{X}')$ be an operator inducing a scalar product $\langle \cdot, \cdot \rangle_{\mathcal{M}} := \langle \mathcal{M} \cdot, \cdot \rangle_{\mathcal{X}' \times \mathcal{X}}$ on $\mathcal{X}$, and similarly $\mathcal{N} \in \mathcal{L}(\mathcal{Y}, \mathcal{Y}')$ on $\mathcal{Y}$. The corresponding induced norms on $\mathcal{X}$ and $\mathcal{Y}$ are denoted by $\|\cdot\|_{\mathcal{M}}$ and $\|\cdot\|_{\mathcal{N}}$. We assume the norm equivalences $\|\cdot\|_{\mathcal{M}} \sim \|\cdot\|_{\mathcal{X}}$ and $\|\cdot\|_{\mathcal{N}} \sim \|\cdot\|_{\mathcal{Y}}$. One immediate example is given by the Riesz operators, e.g. if $\mathcal{M}$ is defined by $\langle \mathcal{M} x, \bar{x} \rangle_{\mathcal{X}' \times \mathcal{X}} := \langle x, \bar{x} \rangle_{\mathcal{X}}$ for all $x, \bar{x} \in \mathcal{X}$. Another example based on the well-known BPX operator will be given in Section 2.3 below. The following theorem defines the concept of the minimal residual discrete solution (see [6, Theorem 3.1] for the proof).

Theorem 2.2. *Let $\mathcal{U} \subseteq \mathcal{X}$ and $\mathcal{V} \subseteq \mathcal{Y}$ be closed subspaces. Let $\mathcal{B}$ be given by (2.8). Assume that the* discrete inf-sup condition

$$\gamma_{\mathcal{U},\mathcal{V}} := \inf_{u \in \mathcal{U} \setminus \{0\}} \sup_{v \in \mathcal{V} \setminus \{0\}} \frac{(\mathcal{B}u)(v)}{\|u\|_{\mathcal{X}} \|v\|_{\mathcal{Y}}} > 0 \tag{2.11}$$

is valid. Then there exists a unique $u \in \mathcal{U}$ satisfying

$$u = \arg\min_{w \in \mathcal{U}} \sup_{v \in \mathcal{V} \setminus \{0\}} \frac{|(\mathcal{B}w - b)(v)|}{\|v\|_{\mathcal{N}}}. \tag{2.12}$$

Moreover, with $x := \mathcal{B}^{-1} b$, the quasi-optimality estimate

$$\|x - u\|_{\mathcal{X}} \leq C \inf_{w \in \mathcal{U}} \|x - w\|_{\mathcal{X}}, \quad \textit{where} \quad C = \frac{\|\mathcal{B}\|_{\mathcal{L}(\mathcal{X},\mathcal{Y})}}{\gamma_{\mathcal{U},\mathcal{V}}} \frac{C_{\mathcal{N}}}{c_{\mathcal{N}}}, \tag{2.13}$$

holds, with the constants of the norm equivalence $c_{\mathcal{N}} \|\cdot\|_{\mathcal{Y}} \leq \|\cdot\|_{\mathcal{N}} \leq C_{\mathcal{N}} \|\cdot\|_{\mathcal{Y}}$.

Assume that we are given finite-dimensional subspaces $\mathcal{U} \subseteq \mathcal{X}$ and $\mathcal{V} \subseteq \mathcal{Y}$, as well as bases $\Phi \subset \mathcal{X}$ for $\mathcal{U}$ and $\Psi \subset \mathcal{Y}$ for $\mathcal{V}$. Assume further that the pair $(\mathcal{U}, \mathcal{V})$ satisfies the *inf-sup condition* (2.11). Set $\mathbb{U} := \mathbb{R}^{\dim \mathcal{U}}$ and $\mathbb{V} := \mathbb{R}^{\dim \mathcal{V}}$. Define the matrices $\mathsf{N} \in \mathbb{V} \times \mathbb{V}$, $\mathsf{B} \in \mathbb{V} \times \mathbb{U}$ and $\mathsf{M} \in \mathbb{U} \times \mathbb{U}$ by

$$\mathsf{N} := \langle \mathcal{N}\Psi, \Psi \rangle_{\mathcal{Y}' \times \mathcal{Y}}, \quad \mathsf{B} := \langle \mathcal{B}\Phi, \Psi \rangle_{\mathcal{Y}' \times \mathcal{Y}}, \quad \text{and} \quad \mathsf{M} := \langle \mathcal{M}\Phi, \Phi \rangle_{\mathcal{X}' \times \mathcal{X}}, \tag{2.14}$$

(i.e., the component B_{ji} is given by $\langle \mathcal{B}\phi_i, \psi_j \rangle_{\mathcal{Y}' \times \mathcal{Y}}$ where $\phi_i \in \Phi$, $\psi_j \in \Psi$) and $\mathbf{b} \in \mathbb{V}$ as the column load vector by $\mathbf{b} := \langle b, \Psi \rangle_{\mathcal{Y}' \times \mathcal{Y}}$. Note that M and N are s.p.d. matrices. We set $\widetilde{\mathsf{B}} := \mathsf{N}^{-\top/2} \mathsf{B} \mathsf{M}^{-1/2}$ and $\widetilde{\mathbf{b}} := \mathsf{N}^{-\top/2} \mathbf{b}$.

Theorem 2.3. *With the above definitions the following hold.*

1. *The condition number $\kappa_2(\widetilde{\mathsf{B}}^\top \widetilde{\mathsf{B}})$ w.r.t. the Euclidean norm satisfies*

$$\kappa_2(\widetilde{\mathsf{B}}^\top \widetilde{\mathsf{B}}) \leq C \tag{2.15}$$

   where $C \geq 0$ is a monotonic function of $\gamma_{\mathcal{U},\mathcal{V}}^{-1} \|\mathcal{B}\|_{\mathcal{L}(\mathcal{X},\mathcal{Y})}$ and the constants in the norm equivalences $\|\cdot\|_{\mathcal{X}} \sim \|\cdot\|_{\mathcal{M}}$ and $\|\cdot\|_{\mathcal{Y}} \sim \|\cdot\|_{\mathcal{N}}$ only.
2. *There exists a unique $\mathbf{u} \in \mathbb{U}$ satisfying*

$$\mathbf{u} = \arg\min_{\mathbf{w} \in \mathbb{U}} \|\mathsf{B}\mathbf{w} - \mathbf{b}\|_{\mathsf{N}^{-1}}, \tag{2.16}$$

   or, equivalently,

$$\widetilde{\mathsf{B}}^\top \widetilde{\mathsf{B}} \widetilde{\mathbf{u}} = \widetilde{\mathsf{B}}^\top \widetilde{\mathbf{b}} \quad \textit{with} \quad \widetilde{\mathbf{u}} := \mathsf{M}^{1/2} \mathbf{u}. \tag{2.17}$$

DOI: 10.1002/nla

3. The function $u := \Phi^\top \mathbf{u} \in \mathcal{U}$ is characterized by (2.12).

For the proof see [6, Proposition 3.2–3.3]. We remark that the first claim of the above theorem implies that M is spectrally equivalent to $\mathsf{B}^\top\mathsf{N}^{-1}\mathsf{B}$, and the equivalence constants depend on the discretization only via the discrete inf-sup constant $\gamma_{\mathcal{U},\mathcal{V}}$. The second states that M is used as a preconditioner for the linear system of equations $\mathsf{B}^\top\mathsf{N}^{-1}\mathsf{B}\mathbf{u} = \mathsf{B}^\top\mathsf{N}^{-1}\mathbf{b}$.

Motivated by the bound (2.15) on the condition number of the "system matrix" $\widetilde{\mathsf{B}}^\top\widetilde{\mathsf{B}}$ we will apply a version of the conjugate gradient method to the preconditioned normal equations (2.17). This will require an efficient method for the computation of (the action of) the inverses of N and M, given in (2.14); we discuss a particular setting where this is possible in the next section.

2.3. The parabolic BPX preconditioner

The BPX preconditioner [12] was developed for second order elliptic partial differential equations, in particular also for anisotropic problems [36], and has been put into greater perspective, see [14, Section 5.4] and references therein. In the optimality of the preconditioner, norm equivalences of the type

$$\|\nu\|_V^2 \sim \sum_{\ell\in\mathbb{N}_0} 2^{2\ell}\|Q_\ell\nu\|_H^2 \quad \forall \nu \in V, \tag{2.18}$$

where $Q_\ell : H \to H$, $\ell \in \mathbb{N}_0$, are suitable projections, play an important role. Starting with such norm equivalences in the spatial domain and in the temporal domain, the preconditioner may be adapted to the parabolic operator, which involves different orders of differentiability. We briefly describe the construction of this "parabolic BPX preconditioner" here and refer to [7] for more details. The first set of requirements is given in the following.

1. In the temporal domain, there exist closed nested subspaces $E_k \subseteq E_{k+1} \subseteq H^1(J)$, $k \in \mathbb{N}_0$, and linear (not necessarily surjective) projections $P_k : H^1(J) \to E_k$, $k \in \mathbb{N}_0$, satisfying

$$\|e\|_{L^2(J)}^2 \sim \sum_{k\in\mathbb{N}_0} \|P_k e\|_{L^2(J)}^2 \quad \forall e \in L^2(J) \tag{2.19}$$

and

$$\|e\|_{H^1(J)}^2 \sim \sum_{k\in\mathbb{N}_0} 2^{2k}\|P_k e\|_{L^2(J)}^2 \quad \forall e \in H^1(J). \tag{2.20}$$

Further, we require that $P_k P_{k'} = 0$ for all nonnegative integers $k \neq k'$.

2. In the spatial domain, there exist closed nested subspaces $V_\ell \subset V$, $\ell \in \mathbb{N}_0$, with $\{0\} =: V_{-1} \subseteq V_\ell \subseteq V_{\ell+1} \subseteq V$ and such that $\bigcup_{\ell\in\mathbb{N}_0} V_\ell$ is dense in V. Further, H-orthogonal projections $Q_{\underline{\ell}} : V \to H$ are needed, with $\underline{\ell}$ ranging in a suitable countable index set (this will be the set of finitely supported multiindices later on), such that

$$d_V\|\nu\|_V^2 \leq \sum_{\underline{\ell}} q_{\underline{\ell}}^2\|Q_{\underline{\ell}}\nu\|_H^2 \leq D_V\|\nu\|_V^2 \quad \forall \nu \in V \tag{2.21}$$

holds with some constants $0 < d_V \leq D_V$ and some $q_{\underline{\ell}} \in \mathbb{R}$, and furthermore, $Q_{\underline{\ell}}Q_{\underline{\ell}'} = 0$ for all $\underline{\ell} \neq \underline{\ell}'$. The connection between $Q_{\underline{\ell}}$ and the subspaces V_ℓ will be given in Section 2.4.2; it is not needed for the theoretical considerations in this section.

For the following recall that, since the duality pairing $\langle\cdot,\cdot\rangle_{V'\times V}$ and the scalar product $\langle\cdot,\cdot\rangle_H$ are compatible, i.e., agree on the set where both are defined, so are $\langle\cdot,\cdot\rangle_{\mathcal{X}'\times\mathcal{X}}$ and $\langle\cdot,\cdot\rangle_{L^2(J;H)}$, as well as $\langle\cdot,\cdot\rangle_{\mathcal{Y}'\times\mathcal{Y}}$ and $\langle\cdot,\cdot\rangle_{[L^2(J;H)\times H]}$.

An important observation is the fact that the norm equivalence (2.21) in V and H-orthogonality of the projections $Q_{\underline{\ell}}$ imply a similar norm equivalence for the dual V'.

DOI: 10.1002/nla

Lemma 2.4. *With (2.21) we also have*

$$D_V^{-1}\|\nu\|_{V'}^2 \le \sum_{\underline{\ell}} q_{\underline{\ell}}^{-2}\|Q_{\underline{\ell}}\nu\|_H^2 \le d_V^{-1}\|\nu\|_{V'}^2 \quad \forall \nu \in V. \tag{2.22}$$

Proof
As in [37, Lemma 1]. □

For any $e \otimes \nu \in \mathcal{X}_0 := \bigcup_{k,\ell\in\mathbb{N}_0} E_k \otimes V_\ell$, we define the operators $\mathcal{M}_+$ and $\mathcal{M}_-$ by

$$\mathcal{M}_\pm(e \otimes \nu) := \sum_{k\in\mathbb{N}_0}\sum_{\underline{\ell}} g_{k,\underline{\ell}}^{\pm 1}(P_k e \otimes Q_{\underline{\ell}}\nu), \quad \text{where} \quad g_{k,\underline{\ell}} := q_{\underline{\ell}}^2 + 2^{2k}q_{\underline{\ell}}^{-2}, \tag{2.23}$$

and their extension to $\mathcal{X}_0$ by linearity. Using (2.19)–(2.20) and (2.21) it can be shown (cf. [36, Proposition 1&2] or [7, Section 6.2]) that

$$\langle \mathcal{M}_+ x, x\rangle_{\mathcal{X}'\times\mathcal{X}} \sim \|x\|_{\mathcal{X}}^2 \quad \forall x \in \mathcal{X}_0 \tag{2.24}$$

holds with constants uniform in x. Moreover, we have $\mathcal{M}_-\mathcal{M}_+ x = x$. Since $\mathcal{X}_0$ is dense in $\mathcal{X}$, the operator $\mathcal{M}_+$ extends uniquely by linearity and continuity to an operator (still denoted by) $\mathcal{M}_+ \in \mathcal{L}(\mathcal{X}, \mathcal{X}')$, and $\mathcal{M}_-$ extends to the inverse thereof, $\mathcal{M}_- = \mathcal{M}_+^{-1} \in \mathcal{L}(\mathcal{X}', \mathcal{X})$.

In order to obtain a pair of subspaces $\mathcal{U}$ and $\mathcal{V}$ satisfying (2.11), we further require closed subspaces $F_k \subset L^2(J)$, $k \in \mathbb{N}_0$, such that $E_k \subseteq F_k$, $k \in \mathbb{N}_0$, and

$$\inf_{k\in\mathbb{N}_0}\inf_{e'\in E_k'\setminus\{0\}}\sup_{f\in F_k\setminus\{0\}}\frac{\langle e', f\rangle_{L^2(J)}}{\|e'\|_{L^2(J)}\|f\|_{L^2(J)}} > 0, \tag{2.25}$$

where $E_k' := \{e' : e \in E_k\}$. For any $(f \otimes \nu, h) \in \mathcal{Y}_0 := \bigcup_{k,\ell\in\mathbb{N}_0}[F_k \otimes V_\ell] \times H$, we then define the operators $\mathcal{N}_+$ and $\mathcal{N}_-$ by

$$\mathcal{N}_\pm(f \otimes \nu, h) := (f \otimes \sum_{\underline{\ell}} q_{\underline{\ell}}^{\pm 2} Q_{\underline{\ell}}\nu, h), \tag{2.26}$$

and the extension to $\mathcal{Y}_0$ by linearity. As above, we have

$$\langle \mathcal{N}_+ y, y\rangle_{\mathcal{Y}'\times\mathcal{Y}} \sim \|y\|_{\mathcal{Y}}^2 \quad \forall y \in \mathcal{Y}_0 \tag{2.27}$$

uniformly in y, and furthermore, $\mathcal{N}_+$ extends uniquely by linearity and continuity to an operator (still denoted by) $\mathcal{N}_+ \in \mathcal{L}(\mathcal{Y}, \mathcal{Y}')$, and $\mathcal{N}_-$ extends to the inverse thereof, $\mathcal{N}_- = \mathcal{N}_+^{-1} \in \mathcal{L}(\mathcal{Y}', \mathcal{Y})$.

Using the notation from Section 2.2, we define the matrices $\mathsf{M}_\pm \in \mathbb{U} \times \mathbb{U}$ and $\mathsf{N}_\pm \in \mathbb{V} \times \mathbb{V}$ by

$$\mathsf{M}_\pm := \langle \mathcal{M}_\pm\Phi, \Phi\rangle_{\mathcal{X}'\times\mathcal{X}} \quad \text{and} \quad \mathsf{N}_\pm := \langle \mathcal{N}_\pm\Psi, \Psi\rangle_{\mathcal{Y}'\times\mathcal{Y}}, \tag{2.28}$$

as well as the *mass matrices* $\mathsf{M}_0 \in \mathbb{U} \times \mathbb{U}$ and $\mathsf{N}_0 \in \mathbb{V} \times \mathbb{V}$ by

$$\mathsf{M}_0 := \langle \Phi, \Phi\rangle_{\mathcal{X}'\times\mathcal{X}} \quad \text{and} \quad \mathsf{N}_0 := \langle \Psi, \Psi\rangle_{\mathcal{Y}'\times\mathcal{Y}}. \tag{2.29}$$

We will apply the results of Section 2.2 with $\mathcal{M} := \mathcal{M}_+$ and $\mathcal{N} := \mathcal{N}_+$; consequently, $\mathsf{M} \equiv \mathsf{M}_+$ and $\mathsf{N} \equiv \mathsf{N}_+$. The following observation will therefore be important for the efficient application of the inverse of the matrices M_+ and N_+ required in the resolution of (2.17).

Proposition 2.5. *There hold*

$$\mathsf{M}_+^{-1} = \mathsf{M}_0^{-1}\mathsf{M}_-\mathsf{M}_0^{-1} \quad \textit{and} \quad \mathsf{N}_+^{-1} = \mathsf{N}_0^{-1}\mathsf{N}_-\mathsf{N}_0^{-1}. \tag{2.30}$$

Proof
We show that $\mathsf{M}_0 = \mathsf{M}_-\mathsf{M}_0^{-1}\mathsf{M}_+$. Indeed, observe that

$$\bar{\mathbf{x}}^\top\mathsf{M}_0\mathbf{x} = \langle \bar{x}, x\rangle_{\mathcal{X}'\times\mathcal{X}} = \langle \mathcal{M}_+^{-1}\bar{x}, \mathcal{M}_+ x\rangle_{\mathcal{X}'\times\mathcal{X}} = \langle \mathcal{M}_-\bar{x}, \mathcal{M}_+ x\rangle_{\mathcal{X}'\times\mathcal{X}} \tag{2.31}$$

$$= (\mathsf{M}_0^{-1}\mathsf{M}_-\bar{\mathbf{x}})^\top\mathsf{M}_0(\mathsf{M}_0^{-1}\mathsf{M}_+\mathbf{x}) = \bar{\mathbf{x}}^\top\mathsf{M}_-\mathsf{M}_0^{-1}\mathsf{M}_+\mathbf{x} \tag{2.32}$$

holds for all $\bar{x} = \Phi^\top\bar{\mathbf{x}}, x = \Phi^\top\mathbf{x} \in \mathcal{U}$. The proof for N_+^{-1} is analogous. □

DOI: 10.1002/nla

2.4. Space-time test and trial spaces

In order to apply the framework of Section 2.2 we construct finite dimensional subspaces $\mathcal{U} \subseteq \mathcal{X}$ and $\mathcal{V} \subseteq \mathcal{Y}$ for which the inf-sup condition (2.11) can be verified. These are constructed as a tensor product of univariate spaces.

2.4.1. Discretization in time We define the *temporal mesh* as

$$\mathcal{T}_k := \{i2^{-(k+1)}T_{\text{final}} : i = 0, \dots, 2^{k+1}\}. \tag{2.33}$$

The space $E_k \subset H^1(J)$ is defined as the space of continuous, $\mathcal{T}_k$-piecewise linear functions on J with the convention $E_{-1} := \{0\}$. Note that $\mathcal{T}_k \subset \mathcal{T}_{k+1}$, $k \in \mathbb{N}_0$, hence the nestedness property $E_k \subset E_{k+1}$, $k \in \mathbb{N}_0$. From [38, Theorem 3.2], or as in [13, Appendix], the properties (2.19)–(2.20) hold with $P_k : H^1(J) \to E_k \cap (E_{k-1})^{\perp_{L^2(J)}}$ being the $L^2(J)$-orthogonal surjective projection, which we assume for P_k from now on.

Starting from the sequence E_k, $k \in \mathbb{N}_0$, we define the spaces F_k, $k \in \mathbb{N}_0$, required for the construction of the operator $\mathcal{N}$ in (2.27), as $F_k := E_{k+1}$. This choice is motivated by the following result, see [6, Proposition 6.1] for the proof.

Proposition 2.6. *With E_k and F_k as above, (2.25) holds.*

2.4.2. Discretization in space As announced in the introduction, we now specialize the discussion to the case

$$V := H^1_0(D) \quad \text{for} \quad D := (-1,1)^d, \tag{2.34}$$

where $d \in \mathbb{N}$. For each $\mu = 1, \dots, d$ and $\ell \in \mathbb{N}_0$, we take $V_\ell^{(\mu)} \subset H^1_0(-1,1)$ as the standard conforming finite element space defined as the space of all continuous, piecewise linear functions w.r.t. the uniform partition of the interval $(-1,1) \subset \mathbb{R}^1$ into $2^{\ell+1}$ subintervals. For $\underline{\ell}_\mu \in \mathbb{N}_0$, $\mu = 1, \dots, d$, we define $V_{\underline{\ell}} := V_{\ell_1}^{(1)} \otimes \cdots \otimes V_{\ell_d}^{(d)} \subset V$. Further, we set $V_L := V_{(L,\dots,L)}$, $L \in \mathbb{N}_0$. For each $\underline{\ell} \in \mathbb{N}_0^d$ we define $Q_{\underline{\ell}} : V \to H$ as the H-orthogonal surjective projection

$$Q_{\underline{\ell}} : V \to V_{\underline{\ell}} \cap \left(\sum_{\underline{\ell} \neq \underline{\ell}' \leq \underline{\ell}} V_{\underline{\ell}'} \right)^{\perp_H},$$

where the sum runs over all $\underline{\ell}'_\mu \in \mathbb{N}_0$, $\mu = 1, \dots, d$, satisfying the constraint, and an empty sum evaluates to the trivial vector space $\{0\} \subset V$. Since (2.21) holds for $d = 1$, see [39, Theorem 5.8], from

$$\|\nu_1 \otimes \cdots \otimes \nu_d\|_V^2 = \sum_{\mu=1}^d \left(\|\nu_\mu\|^2_{H^1_0(-1,1)} \prod_{\mu \neq \mu'=1}^d \|\nu_{\mu'}\|^2_{L^2(-1,1)} \right)$$

it follows (cf. [36, Theorem 3]) that (2.21) holds for $d \geq 1$ with the same constants $0 < d_V \leq D_V < \infty$ for

$$q_{\underline{\ell}} := \sqrt{\sum_{\mu=1}^d 2^{2\underline{\ell}_\mu}}. \tag{2.35}$$

2.4.3. Tensor product space-time spaces One possible construction of finite element spaces $\mathcal{U} \subset \mathcal{X}$ and $\mathcal{V} \subset \mathcal{Y}$ satisfying the inf-sup condition (2.11) can be obtained from the following result.

Theorem 2.7. *With the notation introduced above, there exists $\gamma > 0$ such that the pair*

$$\mathcal{U} := E_K \otimes V_L \quad \text{and} \quad \mathcal{V} := (F_K \otimes V_L) \times V_L \tag{2.36}$$

satisfies $\gamma_{\mathcal{U},\mathcal{V}} \geq \gamma > 0$ for all $K \in \mathbb{N}_0$ and $L \in \mathbb{N}_0$.

DOI: 10.1002/nla

Proof
The claim follows from (2.25) and the following corresponding property of the spaces V_L

$$\inf_{L\in\mathbb{N}_0}\inf_{\nu'\in V_L\setminus\{0\}}\sup_{\nu\in V_L\setminus\{0\}}\frac{\langle\nu',\nu\rangle_{V'\times V}}{\|\nu'\|_{V'}\|\nu\|_V}>0. \tag{2.37}$$

For details we refer to [6, Proof of Theorem 6.3]. □

3. TENSOR FORMAT

With the space-time discretization proposed in Section 2.4, the solution to the parabolic equation is approximated by a vector $\mathbf{x}$, which can be interpreted as a high-dimensional array, or a tensor. The tensor $\mathbf{x}\in\mathbb{R}^{N_t\times N_x\times\cdots\times N_x}$ is of order $d+1$, where N_t is the number of basis functions in time, and N_x the number of basis functions in space. Hence, the storage cost for $\mathbf{x}$ increases exponentially in d. We reduce this cost by approximating $\mathbf{x}$ in a low rank tensor format, described in Section 3.1. We show in Section 3.3 that the matrices B, M_+^{-1} and N_+^{-1} can be efficiently applied to a tensor in such a format. Using a variant of the conjugate gradient method, an approximation of $\mathbf{x}$ in the low rank tensor format can be computed as discussed in Section 3.4.

We now consider a general tensor $\mathbf{x}\in\mathbb{R}^{n_1\times\cdots\times n_d}$ of order $d\in\mathbb{N}$. Low rank formats provide an approximation of $\mathbf{x}$, similar to the truncated SVD for $d=2$. Consider the case of a matrix $\mathbf{x}\in\mathbb{R}^{n_1\times n_2}$. If the matrix $\mathbf{x}$ has a steep singular value decay, it can be approximated by a low rank matrix $UV^\top\approx\mathbf{x}$ with $U\in\mathbb{R}^{n_1\times r}$ and $V\in\mathbb{R}^{n_2\times r}$. Basic operations such as addition, multiplication by a matrix or a scalar and inner product of two low rank matrices $\mathbf{x}=U_\mathbf{x}V_\mathbf{x}^\top$ and $\mathbf{y}=U_\mathbf{y}V_\mathbf{y}^\top$ can be performed efficiently while preserving the low rank structure:

$$\mathbf{x}+\mathbf{y}=U_\mathbf{x}V_\mathbf{x}^\top+U_\mathbf{y}V_\mathbf{y}^\top=[U_\mathbf{x}\;U_\mathbf{y}][V_\mathbf{x}\;V_\mathbf{y}]^\top \tag{3.1}$$

$$A\mathbf{x}=A(UV^\top)=(AU)V^\top,\quad \mathbf{x}B^\top=UV^\top B^\top=U(BV)^\top \tag{3.2}$$

$$\langle\mathbf{x},\mathbf{y}\rangle:=\sum_{i_1=1}^{n_1}\sum_{i_2=1}^{n_2}\mathbf{x}_{i_1,i_2}\mathbf{y}_{i_1,i_2}=\langle U_\mathbf{x}V_\mathbf{x}^\top,U_\mathbf{y}V_\mathbf{y}^\top\rangle=\langle U_\mathbf{x}^\top U_\mathbf{y},V_\mathbf{x}^\top V_\mathbf{y}\rangle. \tag{3.3}$$

Moreover, for a matrix $UV^\top$ of rank r, an approximation with lower rank $\widetilde{r}\le r$ can be efficiently computed using the truncated SVD:

$$U=Q_UR_U,V=Q_VR_V\quad\Rightarrow\quad UV^\top=Q_U(R_UR_V^\top)Q_V^\top=(Q_UX)\Sigma(Q_VY)^\top,$$

where the singular value decomposition $R_UR_V^\top=X\Sigma Y^\top$ was used in the last equation. The best approximation at rank $\widetilde{r}\le r$ w.r.t. the Frobenius norm is obtained by setting the diagonal entries of Σ at the locations $\widetilde{r}+1,\dots,r$ to zero. These operations allow for iterative solution algorithms, such as the Richardson or the conjugate gradient method, to be performed in low rank formats (see Section 3.4).

Let us now consider the general case $d\ge 2$, that is, approximating a tensor $\mathbf{x}\in\mathbb{R}^{n_1\times\cdots\times n_d}$ in a low rank format. The CP and Tucker decompositions are well-known low rank tensor formats, see [15] for a review. Neither of these formats is well suited for our setting. For example, in the Tucker decomposition storage requirements grow exponentially with the order d of the tensor. Therefore, we choose the $\mathcal{H}$-Tucker format [17, 16]. A special case of this format, called Tensor Train format, was independently proposed in [18].

3.1. The hierarchical Tucker format

Here we give a brief overview of the hierarchical Tucker ($\mathcal{H}$-Tucker) format, and refer the reader to [17, 16] for a more detailed explanation.

In order to describe the $\mathcal{H}$-Tucker format, we introduce the concept of *matricization* of a tensor. Consider a splitting of the dimensions into two disjoint sets: $t\dot\cup s=\{1,\dots,d\}$ with $t=\{t_1,\dots,t_k\}$

DOI: 10.1002/nla

and $s = \{s_1, \dots, s_{d-k}\}$. The *matricization* $X^{(t)}$ of a tensor $\mathbf{x}$ with respect to t is obtained by merging the first set t of modes into row indices and the second set s into column indices:

$$X^{(t)} \in \mathbb{R}^{(n_{t_1}\cdots n_{t_k})\times(n_{s_1}\cdots n_{s_{d-k}})} \quad \text{with} \quad (X^{(t)})_{(i_{t_1},\dots,i_{t_k}),(i_{s_1},\dots,i_{s_{d-k}})} := \mathbf{x}_{i_1,\dots,i_d}$$

for any multiindex $(i_1, \dots, i_d) \in \{1, \dots, n_1\} \times \cdots \times \{1, \dots, n_d\}$. In this notation, the *vectorization* $\mathrm{vec}(\mathbf{x})$ is just the matricization with $t = \{1, \dots, d\}$ and $s = \emptyset$. Consider a collection $T \subseteq 2^{\{1,\dots,d\}}$ of subsets $t \subset \{1, \dots, d\}$. We define the *hierarchical rank* $r_t \in \mathbb{N}_0$ for all $t \in T$, and the corresponding set of $\mathcal{H}$-Tucker tensors as

$$\mathcal{H}\text{-Tucker}((r_t)_{t\in T}) := \left\{ \mathbf{x} \in \mathbb{R}^{n_1\times\cdots\times n_d} : \mathrm{rank}(X^{(t)}) \le r_t \ \forall t \in T \right\}. \tag{3.4}$$

For each set $t \in T$, there exist matrices $U_t \in \mathbb{R}^{(n_{t_1}\cdots n_{t_k})\times r_t}$ and $V_t \in \mathbb{R}^{(n_{s_1}\cdots n_{s_{d-k}})\times r_t}$ such that $X^{(t)} = U_t V_t^\top$. The nestedness property $\mathrm{Range}(U_t) \subseteq \mathrm{Range}(U_{t_r} \otimes U_{t_l})$ holds for each disjoint splitting $t_l \dot\cup t_r = t$ [40, Lemma 17], which implies that there exists a matrix B_t such that $U_t = (U_{t_r} \otimes U_{t_l})B_t$. Consequently, it is sufficient to store U_{t_l}, U_{t_r} and B_t to be able to represent U_t. This property, applied recursively, allows a storage-efficient representation of $\mathcal{H}$-Tucker tensors. Consider the example of $d = 4$:

$$\mathrm{vec}(\mathbf{x}) = X^{(1234)} = (U_{34} \otimes U_{12})B_{1234} \tag{3.5}$$
$$U_{12} = (U_2 \otimes U_1)B_{12} \tag{3.6}$$
$$U_{34} = (U_4 \otimes U_3)B_{34} \tag{3.7}$$
$$\Rightarrow \quad \mathrm{vec}(\mathbf{x}) = (U_4 \otimes U_3 \otimes U_2 \otimes U_1)(B_{34} \otimes B_{12})B_{1234}. \tag{3.8}$$

In the $\mathcal{H}$-Tucker format, T is allowed to be any binary tree with the root node $t_{\text{root}} = \{1, \dots, d\}$, leaf nodes t_{leaf} containing only one element, and all other nodes t having exactly two children t_l, t_r with $t_l \dot\cup t_r = t$.

The storage requirements are $\mathcal{O}(dnr + dr^3)$, for $n := \max_{\mu=1,\dots,d} n_\mu$ and $r := \max_{t\in T} r_t$. Addition, multiplication by a matrix and inner product of two tensors in $\mathcal{H}$-Tucker format can be performed efficiently, we refer the reader to [41, 42] for details on the implementation. The *truncation* of a tensor $\mathbf{x}$ in $\mathcal{H}$-Tucker format is the approximation by a tensor $\widetilde{\mathbf{x}}$ with given lower hierarchical ranks $(\widetilde{r}_t)_{t\in T}$, where $\widetilde{r}_t \le r_t$, $t \in T$. An implementation of this operation that requires $\mathcal{O}(dnr^2 + dr^4)$ floating point operations, and satisfies the quasi-optimality property [17]

$$\|\mathbf{x} - \widetilde{\mathbf{x}}\| \le \sqrt{2d-3}\inf\{\|\mathbf{x} - \mathbf{y}\| : \mathbf{y} \in \mathcal{H}\text{-Tucker}((\widetilde{r}_t)_{t\in T})\}, \tag{3.9}$$

is possible. Given two parameters, a *relative truncation accuracy* $\mathsf{rel\_eps} > 0$ and a *maximal truncation rank* $\mathsf{max\_rank} \in \mathbb{N}$, we define the *truncation* $\mathcal{T}$ which chooses the ranks $(\widetilde{r}_t)_{t\in T}$ adaptively such that $\|\mathbf{x} - \widetilde{\mathbf{x}}\| \le \mathsf{rel\_eps}\|\mathbf{x}\|$ if this is possible with $\widetilde{r}_t \le \mathsf{max\_rank}$ for all $t \in T$; otherwise, some of the ranks are set to $\mathsf{max\_rank}$, and the relative accuracy requirement may not be satisfied. For details we refer to [17, 41, 42]. The numerical experiments below are based on the publicly available `htucker` toolbox [41, 42] in MATLAB which provides a representation of a tensor in $\mathcal{H}$-Tucker format, as well as the operations described above.

3.2. Discretized generalized linear least squares problem

We assume that the functions q, p, h are separable, with

$$q(t,\xi) = q_0(t)\prod_{\mu=1}^{d} q_\mu(\xi_\mu), \quad p(t,\xi) = p_0(t)\prod_{\mu=1}^{d} p_\mu(\xi_\mu) \quad \text{and} \quad h(\xi) = \prod_{\mu=1}^{d} h_\mu(\xi_\mu).$$

The extension to finite sums of separable functions is straightforward. Let $K \in \mathbb{N}_0$ and $L \in \mathbb{N}_0$ be fixed and consider the spaces $\mathcal{U}$ and $\mathcal{V}$ defined in (2.36). Let $\nu_i^{(\mu)}$, $i = 1, \dots, 2^L - 1$, be the standard hat functions spanning the space $V_L^{(\mu)}$, $\mu = 1, \dots, d$. Let e_i, $i = 1, \dots, 2^K + 1$, be the

DOI: 10.1002/nla

standard hat functions spanning the space E_K, and similarly f_j, $j = 1, \ldots, 2^{K+1}+1$ for F_K. Then the basis Φ for $\mathcal{U}$ is given by the functions $e^0_{i_0} \otimes \nu^{(1)}_{i_1} \otimes \cdots \otimes \nu^{(d)}_{i_d}$, and the basis Ψ for $\mathcal{V}$ consists of $f^0_{j_0} \otimes \nu^{(1)}_{i_1} \otimes \cdots \otimes \nu^{(d)}_{i_d}$, where $i_0 = 1, \ldots, 2^K+1$, $j_0 = 1, \ldots, 2^{K+1}+1$ and $i_\mu = 1, \ldots, 2^L - 1$, $\mu = 1, \ldots, d$. As described above in Section 2.2, space-time minimal residual Petrov-Galerkin discretization w.r.t. those bases leads to the generalized linear least-squares problem $\arg\min_{\mathbf{u}} \|\mathsf{B}\mathbf{u} - \mathbf{b}\|_{\mathsf{N}^{-1}}$, where the matrix B has the form

$$\begin{pmatrix} C^{(0)} \otimes M^{(1)} \otimes \cdots \otimes M^{(d)} + \sum_{\mu=1}^{d} \widehat{M}^{(0)} \otimes \widehat{M}^{(1)} \otimes \cdots \otimes A^{(\mu)} \otimes \cdots \otimes \widehat{M}^{(d)} \\ (c^{(0)})^\top \otimes M^{(1)} \otimes \cdots \otimes M^{(d)} \end{pmatrix}$$

and

$$\mathbf{b} = \begin{pmatrix} p^{(0)} \otimes p^{(1)} \otimes \cdots \otimes p^{(d)} \\ 1 \otimes h^{(1)} \otimes \cdots \otimes h^{(d)} \end{pmatrix}.$$

Here, and in the following, the superscript in vectors and matrices refers to the mode $\mu = 0, \ldots, d$. These equations can be combined into one Kronecker product structure by using concatenation in the first mode:

$$\mathsf{B} = \widetilde{C}^{(0)} \otimes M^{(1)} \otimes \cdots \otimes M^{(d)} + \sum_{\mu=1}^{d} \widetilde{M}^{(0)} \otimes \widehat{M}^{(1)} \otimes \cdots \otimes A^{(\mu)} \otimes \cdots \otimes \widehat{M}^{(d)} \tag{3.10}$$

$$\mathbf{b} = \widetilde{p}^{(0)} \otimes p^{(1)} \otimes \cdots \otimes p^{(d)} + \widetilde{h}^{(0)} \otimes h^{(1)} \otimes \cdots \otimes h^{(d)}, \tag{3.11}$$

with block matrices $\widetilde{C}^{(0)}$, $\widetilde{M}^{(0)}$ and block vectors $\widetilde{p}^{(0)}$, $\widetilde{h}^{(0)}$, given by

$$\widetilde{C}^{(0)} = \begin{pmatrix} C^{(0)} \\ (c^{(0)})^\top \end{pmatrix}, \ \widetilde{M}^{(0)} = \begin{pmatrix} \widehat{M}^{(0)} \\ 0 \end{pmatrix}, \ \widetilde{p}^{(0)} = \begin{pmatrix} p^{(0)} \\ 0 \end{pmatrix}, \ \widetilde{h}^{(0)} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

The matrices for the spatial domain are defined as follows for $\mu = 1, \ldots, d$:

$$A^{(\mu)}_{ij} = \int_{-1}^{1} q^{(\mu)}(\xi_\mu) \operatorname{grad} \nu^{(\mu)}_i(\xi_\mu) \cdot \operatorname{grad} \nu^{(\mu)}_j(\xi_\mu) \mathrm{d}\xi_\mu, \tag{3.12}$$

$$M^{(\mu)}_{ij} = \int_{-1}^{1} \nu^{(\mu)}_i(\xi_\mu)\nu^{(\mu)}_j(\xi_\mu)\mathrm{d}\xi_\mu, \quad \widehat{M}^{(\mu)}_{ij} = \int_{-1}^{1} q_\mu(\xi_\mu)\nu^{(\mu)}_i(\xi_\mu)\nu^{(\mu)}_j(\xi_\mu)\mathrm{d}\xi_\mu, \tag{3.13}$$

$$p^{(\mu)}_i = \int_{-1}^{1} p_\mu(\xi_\mu)\nu^{(\mu)}_i(\xi_\mu)\mathrm{d}\xi_\mu, \quad h^{(\mu)}_i = \int_{-1}^{1} h_\mu(\xi_\mu)\nu^{(\mu)}_i(\xi_\mu)\mathrm{d}\xi_\mu. \tag{3.14}$$

For the temporal domain, the following matrices are needed:

$$C^{(0)}_{ij} = \int_0^T f_i(t)\frac{\mathrm{d}e_j}{\mathrm{d}t}(t)\mathrm{d}t, \quad \widehat{M}^{(0)}_{ij} = \int_0^T q^{(0)}(t) f_i(t) e_j(t)\mathrm{d}t, \tag{3.15}$$

$$c^{(0)}_j = e_j(0), \quad p^{(0)}_i = \int_0^T p^{(0)}(t) f_i(t)\mathrm{d}t \tag{3.16}$$

$$(M^{(0)}_e)_{ij} = \int_0^T e_i(t)e_j(t)\mathrm{d}t, \quad (M^{(0)}_f)_{ij} = \int_0^T f_i(t)f_j(t)\mathrm{d}t. \tag{3.17}$$

Observe that there are $2^{K+1}+1$ rows, and 2^K+1 columns in $\widetilde{C}^{(0)}$ and $\widetilde{M}^{(0)}$. With this notation, the mass matrices M_0 and N_0 are given by:

$$\mathsf{M}_0 = M^{(0)}_e \otimes M^{(1)} \otimes \cdots \otimes M^{(d)}, \tag{3.18}$$

$$\mathsf{N}_0 = \begin{pmatrix} M^{(0)}_f & 0 \\ 0 & 1 \end{pmatrix} \otimes M^{(1)} \otimes \cdots \otimes M^{(d)} =: \begin{pmatrix} \mathsf{N}_{0,1} & 0 \\ 0 & \mathsf{N}_{0,2} \end{pmatrix}. \tag{3.19}$$

DOI: 10.1002/nla

3.3. Application of matrices to a tensor in low rank format

Note that the application of $A_0 \otimes A_1 \otimes \cdots \otimes A_d$ to a tensor $\mathbf{x}$ in $\mathcal{H}$-Tucker format preserves its hierarchical ranks. Therefore, given a tensor $\mathbf{x}$ in $\mathcal{H}$-Tucker format with hierarchical rank r_t, a straightforward application of B as defined in (3.10) to $\mathbf{x}$ will result in a tensor with hierarchical ranks at most $(d+1)r_t$. This increases the storage cost by a factor of $(d+1)^3$. However, this problem can be alleviated by writing B in the following form (as proposed for the TT format in [18]):

$$\sum_{j_0=1}^{J_0} \cdots \sum_{j_d=1}^{J_d} \mathbf{h}_{j_0,\ldots,j_d} \left(A_{j_0}^{(0)} \otimes \cdots \otimes A_{j_d}^{(d)} \right) \tag{3.20}$$

where $\mathbf{h}$ is a tensor in $\mathcal{H}$-Tucker format with hierarchical ranks s_t. The application of such a matrix to the tensor $\mathbf{x}$ results in a tensor of ranks $k_t \leq r_t s_t$, $t \in T$ [41].

A matrix of "Laplacian" structure, such as the second term in (3.10), can be represented as in (3.20) with hierarchical ranks $s_t = 2$ [18]. Therefore, we can apply B to $\mathbf{x}$ in such a way that the hierarchical ranks increase only by a factor of 3. Incidentally, if the coefficient $q(t,\xi)$ is independent of ξ, i.e., $q(t,\xi) = q(t)$, the matrices $\widehat{M}^{(\mu)}$ and $M^{(\mu)}$ are equal for all μ. It follows that B as a whole has "Laplacian" structure, and can thus be represented with ranks $s_t = 2$.

It is an important observation that the parabolic BPX preconditioners can be written in the form (3.20). To show this, in the following we write $L_0 := K$ and $L_\mu := L$ for $\mu = 1, \ldots, d$. We focus on M, as it is the more involved case. All considerations will apply similarly to N. For $s \in \mathbb{R}$, let us first define

$$\mathsf{M}_s = \sum_{\ell_0=0}^{L_0} \cdots \sum_{\ell_d=0}^{L_d} (\mathbf{g}_{\ell_0,\ldots,\ell_d})^s \left(P_{\ell_0}^{(0)} \otimes \cdots \otimes P_{\ell_d}^{(d)} \right), \tag{3.21}$$

where the tensor $\mathbf{g}$ is given by

$$\mathbf{g}_{\ell_0,\ldots,\ell_d} := \left(\sum_{\mu=1}^{d} 2^{2\ell_\mu} \right) + 2^{2\ell_0} \left(\sum_{\mu=1}^{d} 2^{2\ell_\mu} \right)^{-1}, \tag{3.22}$$

and the projection matrix $P_\ell^{(\mu)}$ is defined for each $\mu = 0, \ldots, d$, $\ell = 1, \ldots, L_\mu$ by (we omit the superscripts $(\cdot)^{(\mu)}$ on matrices in the right hand side)

$$P_\ell^{(\mu)} := M_{L_\mu} \left(S_{L_\mu \backslash \ell} M_\ell^{-1} S_{\ell \swarrow L_\mu} - S_{L_\mu \backslash \ell-1} M_{\ell-1}^{-1} S_{\ell-1 \swarrow L_\mu} \right) M_{L_\mu}, \tag{3.23}$$

$$P_0^{(\mu)} := M_{L_\mu} \left(S_{L_\mu \backslash 0} M_0^{-1} S_{0 \swarrow L_\mu} \right) M_{L_\mu}. \tag{3.24}$$

Here, $S^{(\mu)}_{\ell_2 \backslash \ell_1}$ denotes the prolongation from level ℓ_1 to level ℓ_2 if $\ell_1 \leq \ell_2$, and $S^{(\mu)}_{\ell_1 \swarrow \ell_2} := (S^{(\mu)}_{\ell_2 \backslash \ell_1})^\top$ the reverse operation of restriction, and, furthermore,

$$M_\ell^{(\mu)} := S^{(\mu)}_{\ell \swarrow L_\mu} M^{(\mu)} S^{(\mu)}_{L_\mu \backslash \ell} \quad \mu = 0, \ldots, d, \quad \ell = 0, 1, \ldots, L_\mu, \tag{3.25}$$

where $M^{(\mu)}$ stands for $M_e^{(0)}$ in the case $\mu = 0$. Note that, by definition, for each μ, the symmetric matrices $P_\ell^{(\mu)}$ satisfy

$$P_{\ell'}^{(\mu)} \left(M_{L_\mu}^{(\mu)} \right)^{-1} P_\ell^{(\mu)} = \delta_{\ell'\ell} P_\ell^{(\mu)} \quad \text{for all} \quad \ell', \ell = 0, \ldots, L_\mu \tag{3.26}$$

and

$$\sum_{\ell=0}^{L_\mu} P_\ell^{(\mu)} = M_{L_\mu}^{(\mu)}. \tag{3.27}$$

DOI: 10.1002/nla

		L									
		1	2	3	4	5	6	7	8	9	10
	1	1	1	2	3	5	5	5	5	5	5
	2	1	3	3	6	7	7	7	7	8	8
d	3	1	2	6	7	9	8	8	8	9	10
	4	2	2	5	8	9	8	10	10	10	11
	5	1	2	7	8	8	10	9	11	12	13

Table 3.1. Hierarchical rank of $\mathcal{H}$-Tucker approximation $\widehat{\mathsf{M}}_-$ to the parabolic BPX preconditioner M_- found to yield $J(\mathbf{h},\widehat{\mathbf{h}}) \le 100$, for spatial discretization levels L and spatial dimensions d.

Consequently, owing to (2.23) we have $\mathsf{M}_\pm = \mathsf{M}_{\pm 1}$ and, moreover, the space-time mass matrix M_0 defined in (3.21) coincides with M_s for $s = 0$.

The block diagonal matrix N_- is defined analogously, where the first block has the form

$$\sum_{\ell_1=0}^{L_1} \cdots \sum_{\ell_d=0}^{L_d} \left(\sum_{\mu=1}^{d} 2^{2\ell_\mu} \right) \mathrm{Id} \otimes P_{\ell_1}^{(1)} \otimes \cdots \otimes P_{\ell_d}^{(d)}, \tag{3.28}$$

and the second block is given by $\mathsf{N}_{0,2}^{-1}$.

Recall that we need to compute the action of $\mathsf{M}_+^{-1} = \mathsf{M}_0^{-1}\mathsf{M}_-\mathsf{M}_0^{-1}$ and $\mathsf{N}_+^{-1} = \mathsf{N}_0^{-1}\mathsf{N}_-\mathsf{N}_0^{-1}$ for preconditioning. In order to approximate the action of M_-, we substitute the *entrywise* reciprocal tensor $\mathbf{h}$ of $\mathbf{g}$ by a tensor $\widehat{\mathbf{h}}$ in $\mathcal{H}$-Tucker format, i.e., $\widehat{\mathbf{h}}_{\ell_0,\dots,\ell_d} \approx \mathbf{h}_{\ell_0,\dots,\ell_d} := \mathbf{g}^{-1}_{\ell_0,\dots,\ell_d}$. In our implementation we obtain $\widehat{\mathbf{h}}$ by truncating the *full* tensor $\mathbf{h}$ of size $(K+1)(L+1)^d$. This results in the operators $\widehat{\mathsf{M}}_-$ and $\widehat{\mathsf{N}}_-$. Let us write $\widehat{\mathsf{M}}_+ := \mathsf{M}_0^{-1}\widehat{\mathsf{M}}_-\mathsf{M}_0^{-1}$, and analogously for $\widehat{\mathsf{N}}_+$. As in the case of B, the hierarchical ranks s_t of $\widehat{\mathbf{h}}$ determine the computational cost of applying these matrices to $\mathbf{x}$ in $\mathcal{H}$-Tucker format. Using the variational characterization of the singular values (see (3.13)–(3.14) in [6]), it is easy to check that the operators $\widetilde{\mathsf{B}} := \mathsf{N}_+^{-\top/2}\mathsf{B}\mathsf{M}_+^{-1/2}$ and $\widehat{\mathsf{B}} := \widehat{\mathsf{N}}_+^{-\top/2}\mathsf{B}\widehat{\mathsf{M}}_+^{-1/2}$ satisfy

$$\kappa_2(\widehat{\mathsf{B}}^\top\widehat{\mathsf{B}}) \le \kappa_2(\widehat{\mathsf{M}}_+^{-\top/2}\mathsf{M}_+\widehat{\mathsf{M}}_+^{-1/2})\,\kappa_2(\widehat{\mathsf{N}}_+^{-\top/2}\mathsf{N}_+\widehat{\mathsf{N}}_+^{-1/2})\,\kappa_2(\widetilde{\mathsf{B}}^\top\widetilde{\mathsf{B}}). \tag{3.29}$$

Recall from (2.15) that $\kappa_2(\widetilde{\mathsf{B}}^\top\widetilde{\mathsf{B}})$ is bounded independently of K and L, hence it is enough to choose $\widehat{\mathsf{M}}_+^{-1}$, $\widehat{\mathsf{N}}_+^{-1}$ to be good approximations of M_+^{-1}, N_+^{-1}, respectively. One can show that

$$J(\mathbf{h},\widehat{\mathbf{h}}) := \frac{\max_{\ell_0,\dots,\ell_d} \mathbf{h}_{\ell_0,\dots,\ell_d}/\widehat{\mathbf{h}}_{\ell_0,\dots,\ell_d}}{\min_{\ell_0,\dots,\ell_d} \mathbf{h}_{\ell_0,\dots,\ell_d}/\widehat{\mathbf{h}}_{\ell_0,\dots,\ell_d}} = \kappa_2(\widehat{\mathsf{M}}_+^{-\top/2}\mathsf{M}_+\widehat{\mathsf{M}}_+^{-1/2}), \tag{3.30}$$

thus we need to find a tensor $\widehat{\mathbf{h}}$ in $\mathcal{H}$-Tucker format of small hierarchical ranks s_t such that $J(\mathbf{h},\widehat{\mathbf{h}})$ is small. Unfortunately, we only have access to the quasi-best approximation in Euclidean norm, i.e., $\|\mathbf{h} - \widehat{\mathbf{h}}\|$, which typically leads to large *relative* error in entries that are small in absolute value. This is due to the fact that the relative difference in absolute value of the entries is quite pronounced, and grows exponentially in L. Table 3.1 displays the smallest ranks s_t of $\widehat{\mathbf{h}}$ at which we achieved $J(\mathbf{h},\widehat{\mathbf{h}}) \le 100$, for different choices of d and L, keeping $K = 10$. Note that these are not the minimal ranks needed, but only the minimal ranks available to us while using the quasi-best approximation in the norm $\|\cdot\|$. The ranks go up to 13, which is unacceptably large for our purpose.

In order to alleviate this problem we resort to the approximation

$$\mathsf{M}_- = \mathsf{M}_0\left(\mathsf{M}_0^{-1}\mathsf{M}_{-1/n}\right)^n \approx \mathsf{M}_0\left(\mathcal{T}\circ\mathsf{M}_0^{-1}\mathsf{M}_{-1/n}\right)^n \approx \mathsf{M}_0\left(\mathcal{T}\circ\mathsf{M}_0^{-1}\widehat{\mathsf{M}}_{-1/n}\right)^n, \tag{3.31}$$

where $\mathcal{T}$ represents the truncation to lower hierarchical ranks and $\widehat{\mathsf{M}}_{-1/n}$ results from replacing the full tensor $\mathbf{h}^{1/n} = \mathbf{g}^{-1/n}$ (entrywise power) by a suitable $\mathcal{H}$-Tucker approximation $\widehat{\mathbf{h}^{1/n}}$. The matrix

DOI: 10.1002/nla

		L									
		1	2	3	4	5	6	7	8	9	10
	1	1	1	2	2	3	3	3	3	3	3
	2	1	1	2	3	3	3	3	3	3	4
d	3	1	1	2	3	3	3	3	3	4	4
	4	1	1	2	3	3	3	4	4	4	4
	5	1	1	2	3	3	3	4	4	4	4

Table 3.2. Hierarchical rank of $\mathcal{H}$-Tucker approximation $\widehat{\mathsf{M}}_{-1/n}$ to the parabolic BPX preconditioner $\mathsf{M}_{-1/n}$ found to yield $J(\mathbf{h}^{1/n}, \widehat{\mathbf{h}^{1/n}})^n \leq 100$, for spatial discretization levels L and spatial dimensions d with $n = 4$.

$\widehat{\mathsf{M}}_{-1/n}$ thus depends on the $\mathcal{H}$-Tucker approximation of $\mathbf{h}^{1/n}$, which now has a less pronounced relative difference in entry sizes. Indeed, Table 3.2 shows the hierarchical ranks as described for Table 3.1, for the condition $J(\mathbf{h}^{1/n}, \widehat{\mathbf{h}^{1/n}})^n \leq 100$, for $n = 4$. The hierarchical ranks s_t are now significantly smaller. Since these enter as $n(s_t)^4$ into the computation cost, this is a dramatic improvement. Numerical experiments in Section 4 show that this approach is indeed profitable.

As an alternative to full pointwise inversion of $\mathbf{g}$ with subsequent truncation, we investigate the approximation of the tensor $\mathbf{h}$ by means of *black box approximation* [43, 44, 45, 46, 47]. Given a function which returns any entry of the required tensor, these methods heuristically construct a low-rank approximation of that tensor. In the following numerical experiment, we use a MATLAB implementation [48], of the algorithm described in [43, 44].

Figure 3.1 shows the results of applying this method to the tensor $\mathbf{h}^{1/n}$, with $d = 5$, $K = L = 10$, $n = 4$, and with different maximal ranks for the resulting $\mathcal{H}$-Tucker approximation $\widehat{\mathbf{h}^{1/n}}$. The relative error in 2-norm, which the truncation and black box methods both aim to minimize, is shown in Figure 3.1 (left). Relevant to our application, however, is the impact on the condition number, represented by $J(\mathbf{h}^{1/n}, \widehat{\mathbf{h}^{1/n}})^n$, see (3.30), shown in the right part of Figure 3.1. Applying $\mathcal{H}$-Tucker truncation results in the blue lines in Figure 3.1, while the black box method results in the black lines. Note that the black box has a randomized component, and thus the error is not strictly descending with the rank.

The condition number $J(\mathbf{h}^{1/n}, \widehat{\mathbf{h}^{1/n}})^n$ is too large for the moderate ranks of 3-4 that we can afford in the CGNR algorithm. However, note that the 2-norm error for rank 15 is quite small. We apply $\mathcal{H}$-Tucker truncation to the tensor resulting from the black box method for rank 15, which gives us the red lines in Figure 3.1. This results in a tensor of rank 4 such that the condition number J is smaller than 100, just as in the case of direct truncation.

This combined method can be used to construct $\widehat{\mathbf{h}^{1/n}}$ for higher dimensions d, where we cannot directly store $\mathbf{h}^{1/n}$. However, since the method requires additional parameter choices (initial rank in the black box method, and the rank for subsequent truncation), we will not use it in the following numerical experiments.

Another alternative to pointwise inversion with subsequent truncation is the Newton-Schulz iteration, proposed for the Tensor-Train low rank format in [18].

3.4. The preconditioned conjugate gradient method

Low-rank tensor variants of classical iterative methods for linear systems have been recently proposed in [19, 20, 21]. A classical iterative method (e.g. Richardson, conjugate gradient) is formulated with low rank tensors as its iterates. Due to repeated addition and application of matrices to the iterates, the ranks will grow rapidly throughout the iteration. This growth is limited by truncation $\mathcal{T} : \mathbf{u} \mapsto \widetilde{\mathbf{u}}$ of the iterates with certain relative accuracy rel_eps and maximal truncation rank max_rank.

DOI: 10.1002/nla

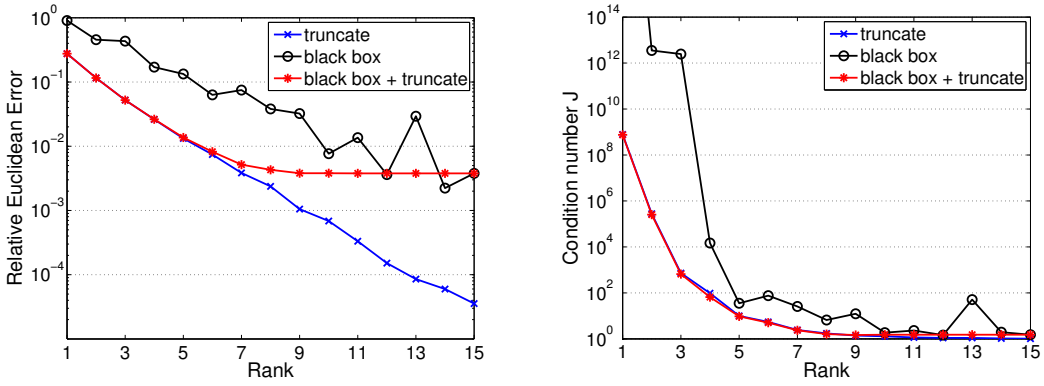

Figure 3.1. Comparison of direct pointwise inversion with subsequent truncation, black box approximation, and black box approximation with subsequent truncation, for $d = 5$, $K = L = 10$, $n = 4$.
Left: Relative euclidean error $\|\mathbf{h}^{1/n} - \widehat{\mathbf{h}^{1/n}}\|/\|\mathbf{h}^{1/n}\|$, **right**: Condition number $J(\mathbf{h}^{1/n}, \widehat{\mathbf{h}^{1/n}})^n$.

We apply the conjugate gradient (CG) method to $\mathsf{B}^\top \mathsf{N}^{-1}\mathsf{B}\mathbf{u} = \mathsf{B}^\top \mathsf{N}^{-1}\mathbf{b}$ with the preconditioner M^{-1}. We reformulate the CG method for the preconditioned normal equations to resemble the CGNR method [49]. The matrices N^{-1}, M^{-1} in Algorithm 1 denote the (approximate) preconditioners described in Section 3.4. We monitor the following residuals of the k-th iterate: the *CG residual* $\|\mathsf{B}^\top \mathsf{N}^{-1}\mathsf{B}\mathbf{u}_k - \mathsf{B}^\top \mathsf{N}^{-1}\mathbf{b}\|_{\mathsf{M}^{-1}}$ and the *least squares residual* $\|\mathsf{B}\mathbf{u}_k - \mathbf{b}\|_{\mathsf{N}^{-1}}$.

Typically, the hierarchical ranks grow in the transient phase of the iteration, and decrease as the iterates approach the least squares minimizer. It has been observed in [21] that a good preconditioner is essential for such low rank variants of classical iterative solvers, in order to keep the ranks of the iterates moderate without significant loss of accuracy.

As a measure of the error of the discrete solution $\mathbf{u}$ we will use the *estimated error*, computed as the least squares residual $\|\mathsf{B}\widehat{\mathbf{u}} - \mathbf{b}\|_{\mathsf{N}^{-1}}$ for the prolongation of the solution $\mathbf{u}$ onto level $K = L = 11$ (the load vector $\mathbf{b}$ is also computed in this resolution), denoted by $\widehat{\mathbf{u}}$, where the matrix N^{-1} is approximated using $n = 4$ and $s_t \equiv 5$, $\mathsf{rel\_eps} = 10^{-8}$ and $\mathsf{max\_rank} = 40$ (see Section 3.3).

Algorithm 1 Variant of low rank CGNR method in $\mathcal{H}$-Tucker format

Input: Functions applying $\mathcal{T} \circ \mathsf{B}$, $\mathcal{T} \circ \mathsf{M}^{-1}$, $\mathcal{T} \circ \mathsf{N}^{-1}$ to a tensor in $\mathcal{H}$-Tucker format, right-hand side $\mathbf{b}$ in $\mathcal{H}$-Tucker format. Truncation operator $\mathcal{T}$ with rel. accuracy ϵ_{rel}.
Output: Tensor $\mathbf{u}$ fulfilling $\|\mathsf{B}\mathbf{u} - \mathbf{b}\|_{\mathsf{N}^{-1}} \leq \mathsf{tol}$.

$\mathbf{u}_0 = 0$, $\mathbf{r}_0 = \mathbf{b}$, $\mathbf{s}_0 = \mathsf{B}^\top \mathsf{N}^{-1}\mathbf{r}_0$, $\mathbf{p}_0 = \mathbf{s}_0$, $\gamma_0 = \langle \mathsf{B}\mathbf{p}_0, \mathsf{B}\mathbf{p}_0\rangle_{\mathsf{N}^{-1}}$, $k = 0$
while $\|\mathbf{r}_k\|_{\mathsf{N}^{-1}} > \mathrm{tol}$ **do**
  $\alpha_k = \langle \mathbf{s}_k, \mathbf{p}_k\rangle/\gamma_k$
  $\mathbf{u}_{k+1} = \mathbf{u}_k + \alpha_k \mathbf{p}_k$ $\quad$ $\mathbf{u}_{k+1} \leftarrow \mathcal{T}(\mathbf{u}_{k+1})$
  $\mathbf{r}_{k+1} = \mathbf{b} - \mathsf{B}\mathbf{u}_{k+1}$ $\quad$ $\mathbf{r}_{k+1} \leftarrow \mathcal{T}(\mathbf{r}_{k+1})$
  $\mathbf{s}_{k+1} = \mathsf{B}^\top \mathsf{N}^{-1}\mathbf{r}_{k+1}$
  $\mathbf{z}_{k+1} = \mathsf{M}^{-1}\mathbf{s}_{k+1}$
  $\beta_{k+1} = -\langle \mathsf{B}\mathbf{z}_{k+1}, \mathsf{B}\mathbf{p}_k\rangle_{\mathsf{N}^{-1}}/\gamma_k$
  $\mathbf{p}_{k+1} = \mathbf{z}_{k+1} + \beta_{k+1}\mathbf{p}_k$ $\quad$ $\mathbf{p}_{k+1} \leftarrow \mathcal{T}(\mathbf{p}_{k+1})$
  $\gamma_{k+1} = \langle \mathsf{B}\mathbf{p}_{k+1}, \mathsf{B}\mathbf{p}_{k+1}\rangle_{\mathsf{N}^{-1}}$
  $k = k + 1$
end while
$\mathbf{u} = \mathbf{u}_k$

DOI: 10.1002/nla

4. NUMERICAL EXPERIMENTS

In our numerical experiments, we set $K = 8$ and $L = 8$ for the discretization levels in time and space, resp., cf. Section 2.4, unless specified otherwise. Hence, the trial space is spanned by 513×511^d functions which are tensor products of standard univariate hat functions. In addition to the isotropic case $q : J \times D \to \mathbb{R}$ discussed so far, we investigate the anisotropic case $q : J \times D \to \mathbb{R}^{d\times d}$ with q taking values in the set of diagonal and positive definite matrices such that $0 < a_{\min} \le q_{ii}(t, \xi) \le a_{\max} < \infty$ still holds, cf. (2.4). We set $T_{\text{final}} = 2$, $D = (-1, 1)^d$ with $d = 1, \dots, 5$, as well as $h \equiv 0$ and

$$p(t, \xi) = \left(1 + \sin \frac{\pi t}{2}\right) \prod_{j=1}^{d} (1 - \xi_j^2) e^{\xi_j}, \quad t \in J, \quad \xi \in D.$$

In all cases, the preconditioners are approximated with $n = 4$ and $s_t = 3$, see Section 3.3. All numerical experiments have been performed in MATLAB version 7.7.0.471, on an Intel Xeon DP X5450 with 3 GHz and 2×6MB L2 Cache.

4.1. *Isotropic diffusion*

Throughout this section we set $q \equiv 1$. Figure 4.1 shows the typical convergence behavior of the CG method. Here, $d = 2$, $K = L = 5$, $\mathsf{max\_rank} = 50$ and we compare relative truncation accuracy $\mathsf{rel\_eps} = 10^{-2}$ and $\mathsf{rel\_eps} = 10^{-6}$. Evidently, for $\mathsf{rel\_eps} = 10^{-2}$, the hierarchical ranks remain very low (below 5), and the truncation error does not allow the CG residual to decrease below 10^{-4}. Therefore, the least squares residual also stagnates. For $\mathsf{rel\_eps} = 10^{-6}$, the convergence follows the typical pattern: the least squares residual clearly stagnates while the CG residual still decreases; the stagnation is therefore not caused by the truncation (but rather by the insufficiently exact discretization). The ranks (depicted is $\max_{t\in T} r_t$ of the iterate $\mathbf{u}_k$) grow in the transient phase of the CG method but eventually decrease to about 8. A good preconditioner is therefore required to limit the number of iterations and consequently the intermediate ranks and the computational effort.

The effect of the choice of the truncation rank in the preconditioner during the CG iteration on the solution time is displayed in Figure 4.2 (left). We observe that rank 1 does not yield an effective preconditioner; rank 5 does improve the quality of the preconditioner over rank 3, but the overhead during each application of the preconditioner does not pay off. In Figure 4.2 (right), we show the estimated error for different spatial and temporal discretization levels $L, K \in \{4, 8\}$ during the CG iteration, when all other parameters are generously chosen (for instance, rank 5 for the preconditioner). In Figure 4.3, we show the estimated error as a function of the spatial and as a function of the temporal mesh width, when all other parameters are generously chosen. In both cases, we observe the expected rate of convergence of approximately one with respect to the mesh width.

Now we set $K = L = 8$, and vary the spatial dimension d. Figure 4.4 shows the convergence history of the least squares residual as a function of the iteration count and of the computation time. We use $\mathsf{rel\_eps} = 10^{-5}$ and $\mathsf{max\_rank} = 30$ for $d = 1, \dots, 5$. The CG convergence rate deteriorates with the number of spatial dimensions d. Figure 4.5 shows the (wall-clock) time until the least squares residual falls below 2×10^{-3} and the average time per iteration as a function of the spatial dimension $d = 1, \dots, 5$.

4.2. *Anisotropic diffusion*

Let $\gamma_1 := 1$, $\gamma_2 := 10^{-1}$, $\gamma_3 := 10^{-2}$, etc. Let $q : J \times D \to \mathbb{R}^{d\times d}$ be the constant function equal to the diagonal matrix $q \equiv \operatorname{diag}(\gamma_1, \dots, \gamma_d)$. Consider now (2.1)–(2.3) with q so defined. In order to take the anisotropy into account, we modify $\mathbf{g}_{\ell_0,\dots,\ell_d}$ in the definition (3.21) of the parabolic BPX

DOI: 10.1002/nla

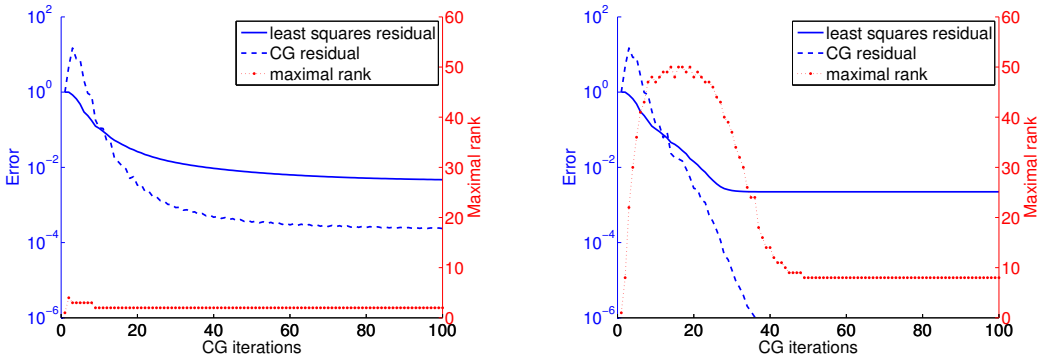

Figure 4.1. Convergence plot of CG, $K = L = 5$. Truncation tolerance: 10^{-2} (**left**), 10^{-6} (**right**). Isotropic case ($d = 2$) of Section 4.1.

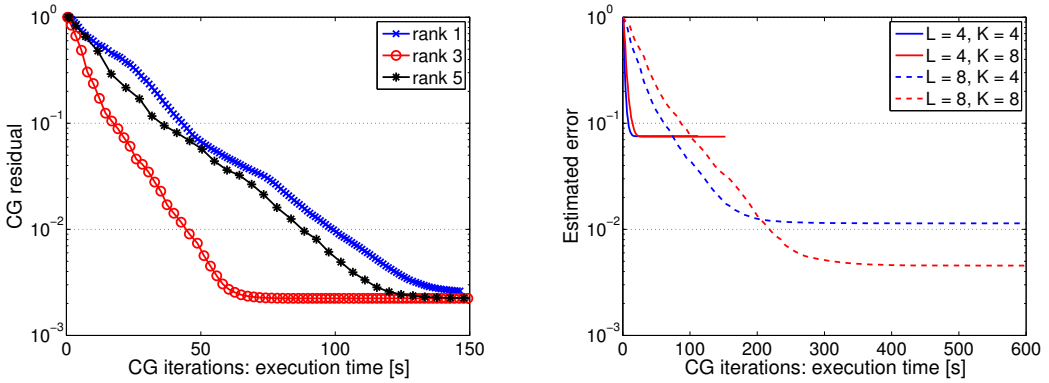

Figure 4.2. CG residual versus CG iteration time for different truncation ranks in the parabolic BPX preconditioner with discretization levels $K = L = 5$ (**left**). Estimated error of the discrete solution for discretization levels $K, L \in \{4, 8\}$ versus CG iteration time (**right**). Isotropic case ($d = 2$) of Section 4.1.

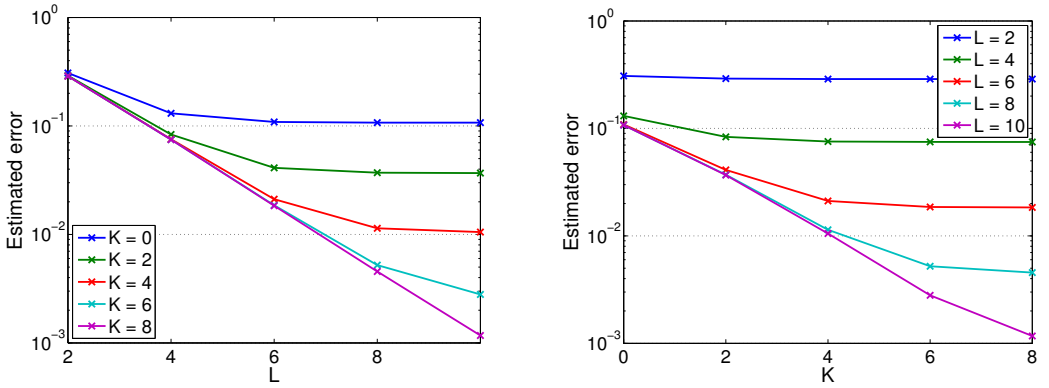

Figure 4.3. Estimated error of the discrete solution for various levels of spatial and temporal discretization L and K. The rate is approximately one in the mesh width 2^{-L} (if K is large enough) and 2^{-K} (if L is large enough), respectively, as can be expected from the approximation properties. Isotropic case ($d = 2$) of Section 4.1.

preconditioner as follows,

$$\mathbf{g}_{\ell_0,\ldots,\ell_d} = \left(\sum_{\mu=1}^{d} \gamma_\mu 2^{2\ell_\mu}\right) + 2^{2\ell_0}\left(\sum_{\mu=1}^{d} \gamma_\mu 2^{2\ell_\mu}\right)^{-1}, \tag{4.1}$$

DOI: 10.1002/nla

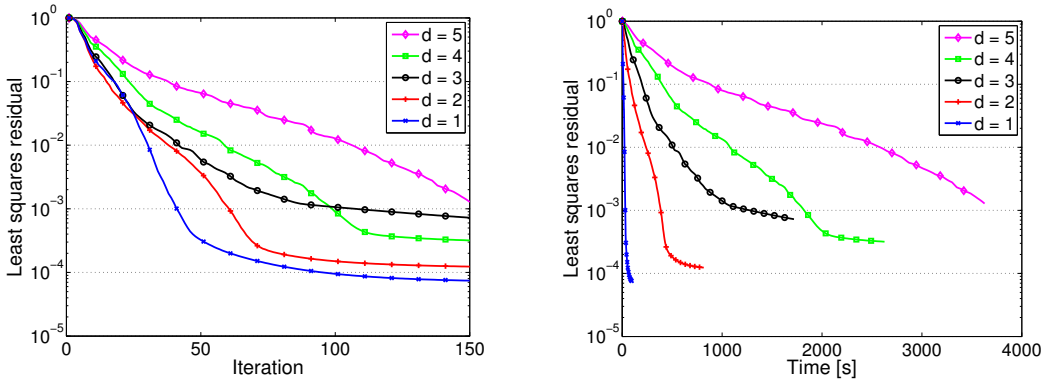

Figure 4.4. Convergence of the CG method for different dimensions d. Isotropic case of Section 4.1.

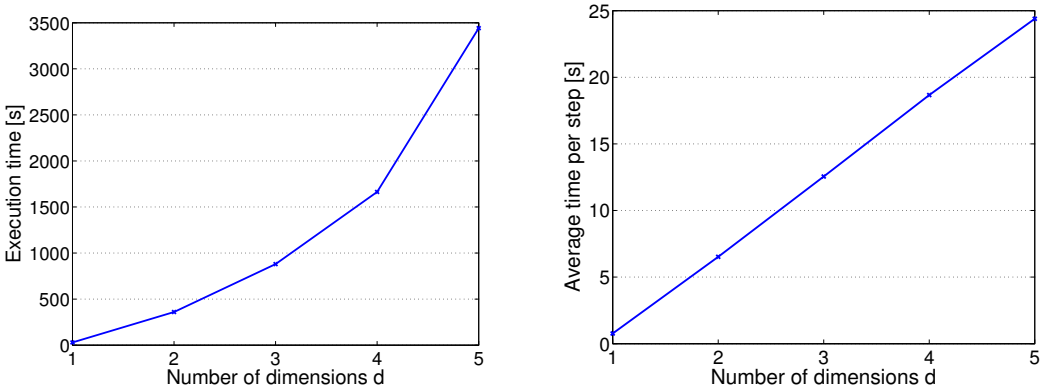

Figure 4.5. Timings for the CG method until a least squares residual of 2×10^{-3} is reached. Isotropic case of Section 4.1.

and similarly for N. Note that this implies a modification of the norms that are used to measure the residuals. The following experiments are identical to the ones in the last section, and the conclusions for Figure 4.6–4.7 correspond to those of Figure 4.4–4.5. The convergence rate, however, now deteriorates only moderately in dependence on the dimension d, as opposed to the isotropic case. In fact, we observe close to linear scaling in the dimension $d = 1, \dots, 5$ for the computation time (total, as well as per iteration).

5. CONCLUSIONS

This paper merges a priori stable minimal residual Petrov-Galerkin space-time discretizations and adaptive low rank approximation of high-dimensional tensors in the hierarchical Tucker format for the solution of parabolic evolution equations, focussing on the case of a symmetric generator such as the Laplace operator.

The minimal residual Petrov-Galerkin discretization of the model diffusion equation yields a linear system of the form $\mathsf{B}^\top \mathsf{N}^{-1} \mathsf{B} \mathbf{u} = \mathsf{B}^\top \mathsf{N}^{-1} \mathbf{b}$ along with a preconditioner M. The matrices N and M are based on an extension of the elliptic BPX preconditioner into the space-time setting and render the linear system well-conditioned uniformly in the discretization level, at least if applied exactly. This is crucial for the performance of the low rank variant of the conjugate gradient method: a well-conditioned system is required to maintain moderate ranks of the iterates, and consequently, low requirements on both, storage and computational time. The tensor structure of the system matrix

DOI: 10.1002/nla

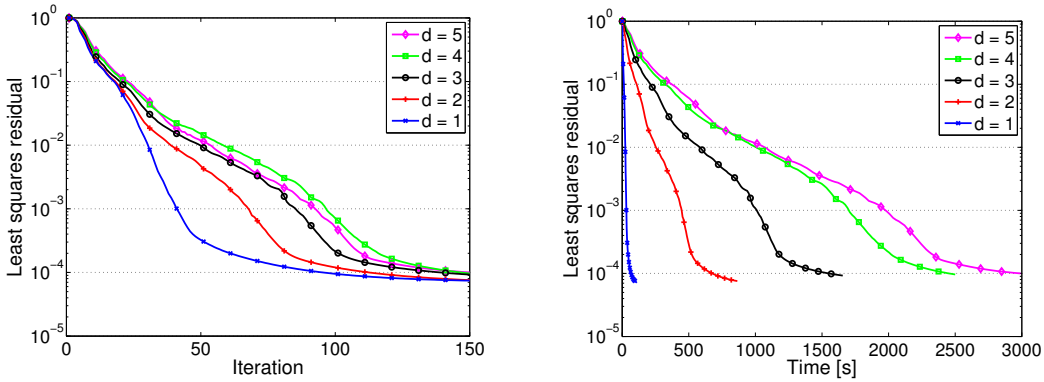

Figure 4.6. Convergence of the CG method for different dimensions d. Anisotropic case of Section 4.2.

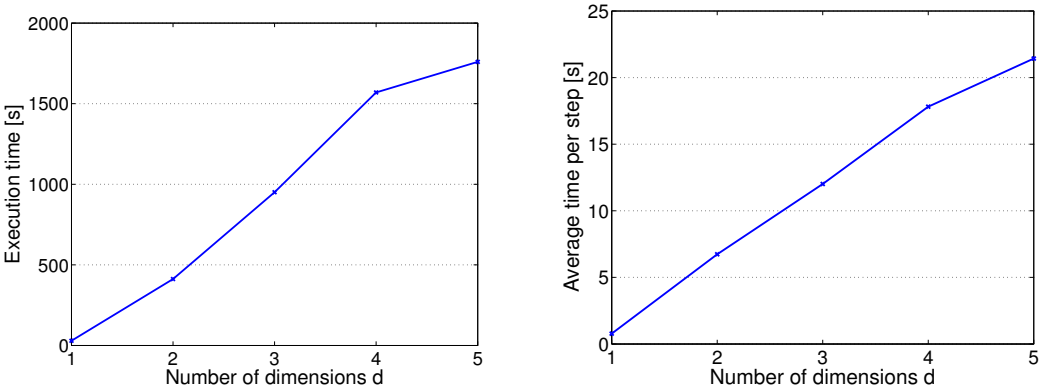

Figure 4.7. Timings for the CG method until a least squares residual of 2×10^{-3} is reached. Anisotropic case of Section 4.2.

(in the case of separable input data) and of the "parabolic BPX preconditioners" proposed here is very convenient: they can be approximated by means of the hierarchical Tucker format and applied to tensors in this format, in particular exploiting parallel architectures. Our numerical experiments demonstrate the potential of this combined approach. In particular, for the anisotropic diffusion, where the conductivity coefficient along the dimension μ decays exponentially in μ, linear scaling of the computational time in the number of spatial dimensions $d = 1, \ldots, 5$ was achieved.

The authors acknowledge the support and the useful suggestions of D. Kressner, Ch. Schwab, and of the anonymous referees. The revision of this manuscript has greatly profited from the first author's visit to EPF Lausanne, supported by D. Kressner.

REFERENCES

1. Thomée V. *Galerkin finite element methods for parabolic problems*, vol. 25. 2nd edn., Springer-Verlag: Berlin, 2006.
2. Gander MJ, Vandewalle S. Analysis of the parareal time-parallel time-integration method. *SIAM J. Sci. Comput.* 2007; **29**(2):556–578.
3. Lions JL, Magenes E. *Non-homogeneous boundary value problems and applications. Vol. I.* Springer-Verlag: New York, 1972.
4. Schwab C, Stevenson R. Space-time adaptive wavelet methods for parabolic evolution problems. *Math. Comp.* 2009; **78**(267):1293–1318.
5. Chegini N, Stevenson R. Adaptive wavelet schemes for parabolic problems: Sparse matrices and numerical results. *SIAM J. Numer. Anal.* 2011; **49**(1):182–212.

DOI: 10.1002/nla

6. Andreev R. Stability of sparse space–time finite element discretizations of linear parabolic evolution equations. *IMA Journal of Numerical Analysis* 2013; **33**(1):242–260.
7. Andreev R. Stability of space-time Petrov-Galerkin discretizations for parabolic evolution equations. PhD Thesis, ETH Zürich 2012. ETH Diss. No. 20842.
8. Andreev R. Space-time discretization of the heat equation. *Numer Algorithms* 2014; **(online)**.
9. Cohen A, Dahmen W, DeVore R. Adaptive wavelet methods for elliptic operator equations: convergence rates. *Math. Comp.* 2001; **70**(233):27–75 (electronic).
10. Cohen A, Dahmen W, DeVore R. Adaptive wavelet methods. II. Beyond the elliptic case. *Found. Comput. Math.* 2002; **2**(3):203–245.
11. Kestler S, Steih K, Urban K. An efficient space-time adaptive wavelet Galerkin method for time-periodic parabolic partial differential equations. *ArXiv e-prints* 2014; .
12. Bramble JH, Pasciak JE, Xu J. Parallel multilevel preconditioners. *Math. Comp.* 1990; **55**(191):1–22.
13. Xu J. Iterative methods by space decomposition and subspace correction. *SIAM Rev.* 1992; **34**(4):581–613.
14. Xu J. The method of subspace corrections. *J. Comput. Appl. Math.* 2001; **128**(1-2):335–362.
15. Kolda TG, Bader BW. Tensor decompositions and applications. *SIAM Review* 2009; **51**(3):455–500.
16. Hackbusch W, Kühn S. A new scheme for the tensor representation. *J. Fourier Anal. Appl.* 2009; **15**(5):706–722.
17. Grasedyck L. Hierarchical Singular Value Decomposition of Tensors. *SIAM J. Matrix Anal. Appl.* 2010; **31**(4):2029–2054.
18. Oseledets IV. Tensor-train decomposition. *SIAM J. Sci. Comput.* 2011; **33**(5):2295–2317.
19. Ballani J, Grasedyck L. A projection method to solve linear systems in tensor format. *Num. Lin. Alg. App.* 2013; **20**(1):27–43.
20. Khoromskij BN, Schwab C. Tensor-Structured Galerkin Approximation of Parametric and Stochastic Elliptic PDEs. *SIAM J. Sci. Comput.* 2011; **33**(1):364–385.
21. Kressner D, Tobler C. Low-rank tensor Krylov subspace methods for parametrized linear systems. *SIAM J. Matrix Anal. Appl.* 2011; **32**(4):1288–1316.
22. Khoromskij BN. Tensor-structured preconditioners and approximate inverse of elliptic operators in $\mathbb{B}R^d$. *Constr. Approx.* 2009; **30**(3):599–620.
23. Oseledets I, Dolgov S. Solution of linear systems and matrix inversion in the TT-format. *SIAM Journal on Scientific Computing* 2012; **34**(5):A2718–A2739.
24. Holtz S, Rohwedder T, Schneider R. The alternating linear scheme for tensor optimization in the Tensor Train format. *SIAM J. Sci. Comput.* 2012; **34**(2):A683–A713.
25. Khoromskij BN, Oseledets IV. Quantics-TT collocation approximation of parameter-dependent and stochastic elliptic PDEs. *Comp. Meth. in Applied Math.* 2010; **10**(4):376–394.
26. Dolgov S, Khoromskij B. Simultaneous state-time approximation of the chemical master equation using tensor product formats. *ArXiv e-prints* 2013; .
27. Kazeev V, Khammash M, Nip M, Schwab C. Direct solution of the chemical master equation using quantized tensor trains. *Technical Report 2013-04*, Seminar for Applied Mathematics, ETH Zürich, Switzerland 2013.
28. Khoromskij BN. Tensors-structured Numerical Methods in Scientific Computing: Survey on Recent Advances. *Chemometr. Intell. Lab. Syst.* 2012; **110**:1–19.
29. Gavrilyuk IP, Khoromskij BN. Quantized-TT-Cayley transform to compute dynamics and spectrum of high-dimensional Hamiltonians. *Comp. Meth. in Applied Math.* 2011; **11**(3):273–290.
30. Arnold A, Jahnke T. On the approximation of high-dimensional differential equations in the hierarchical Tucker format. *BIT* 2014 (online); .
31. Lubich C, Rohwedder T, Schneider R, Vandereycken B. Dynamical approximation of hierarchical Tucker and tensor-train tensors. *SIAM J. Matrix Anal. Appl.* 2013; **34**(2):470–494.
32. Hiptmair R. Operator preconditioning. *Comput. Math. Appl.* 2006; **52**(5):699–706.
33. Bornemann F, Yserentant H. A basic norm equivalence for the theory of multilevel methods. *Numer. Math.* 1993; **64**(4):455–476.
34. Reed M, Simon B. *Methods of modern mathematical physics. I. Functional analysis*. Academic Press: New York, 1972.
35. Fattorini HO. *Infinite dimensional linear control systems*. Elsevier Science B.V.: Amsterdam, 2005.
36. Griebel M, Oswald P. Tensor product type subspace splittings and multilevel iterative methods for anisotropic problems. *Adv. Comput. Math.* 1995; **4**(1-2):171–206.
37. Oswald P. Multilevel norms for $H^{-1/2}$. *Computing* 1998; **61**(3):235–255.
38. Dahmen W, Kunoth A. Multilevel preconditioning. *Numer. Math.* 1992; **63**(3):315–344.
39. Dahmen W. *Wavelet and multiscale methods for operator equations*, *Acta Numer*, vol. 6. Cambridge Univ. Press: Cambridge, 1997; 55–228.
40. Grasedyck L. Hierarchical Low Rank Approximation of Tensors and Multivariate Functions 2010. Lecture notes of Zürich summer school on Sparse Tensor Discretizations of High-Dimensional Problems.
41. Kressner D, Tobler C. `htucker` – a MATLAB toolbox for tensors in hierarchical Tucker format. *ACM Transactions on Mathematical Software* 2014 (to appear); Available at `http://anchp.epfl.ch`.
42. Tobler C. Low rank tensor methods for linear systems and eigenvalue problems. PhD Thesis, ETH Zürich 2012. Diss ETH No. 20320.
43. Ballani J. Fast evaluation of near-field boundary integrals using tensor approximations. PhD Thesis, Universität Leipzig 2012.
44. Ballani J, Grasedyck L, Kluge M. Black box approximation of tensors in hierarchical Tucker format. *Linear Algebra Appl.* 2013; **438**(2):639–657.
45. Oseledets IV, Savostyanov DV, Tyrtyshnikov EE. Cross approximation in tensor electron density computations. *Numer. Linear Algebra Appl* 2010; **17**(6):935–952.
46. Oseledets IV, Tyrtyshnikov EE. TT-cross approximation for multidimensional arrays. *Linear Algebra Appl.* 2010; **432**(1):70–88.

47. Savostyanov DV, Oseledets IV. Fast adaptive interpolation of multi-dimensional arrays in tensor train format. *Proceedings of 7th International Workshop on Multidimensional Systems (nDS)*, IEEE, 2011.
48. Tobler C. Black box approximation of a tensor in HTD 2013. MATLAB code implementing a method described by Ballani, Grasedyck and Kluge. Available at http://anchp.epfl.ch/htucker.
49. Saad Y. *Iterative Methods for Sparse Linear Systems*. 2nd edn., SIAM: Philadelpha, PA, 2003.

DOI: 10.1002/nla