TECHNISCHE UNIVERSITÄT BERLIN

Tensor Networks and Hierarchical Tensors for the Solution of High-Dimensional Partial Differential Equations

Markus Bachmayr **Reinhold Schneider**
André Uschmajew

Preprint 2015/28
Preprint-Reihe des Instituts für Mathematik
Technische Universität Berlin
`http://www.math.tu-berlin.de/preprints`

Preprint 2015/28 **November 2015**

Tensor Networks and Hierarchical Tensors for the Solution of High-dimensional Partial Differential Equations

Markus Bachmayr · **Reinhold Schneider** · **André Uschmajew**

Abstract Hierarchical tensors can be regarded as a generalisation, preserving many crucial features, of the singular value decomposition to higher-order tensors. For a given tensor product space, a recursive decomposition of the set of coordinates into a dimension tree gives a hierarchy of nested subspaces and corresponding nested bases. The dimensions of these subspaces yield a notion of multilinear rank. This rank tuple, as well as quasi-optimal low-rank approximations by rank truncation, can be obtained by a hierarchical singular value decomposition. For fixed multilinear ranks, the storage and operation complexity of these hierarchical representations scale only linearly in the order of the tensor. As in the matrix case, the set of hierarchical tensors of a given multilinear rank is not a convex set, but forms an open smooth manifold. A number of techniques for the computation of low-rank approximations have been developed, including local optimisation techniques on Riemannian manifolds as well as truncated iteration methods, which can be applied for solving high-dimensional partial differential equations. In a number of important cases, quasi-optimality of approximation ranks and computational complexity have been analysed. This article gives a survey of these developments. We also discuss applications to problems in uncertainty quantification, to the solution of the electronic Schrödinger equation in the

M.B. was supported by ERC AdG BREAD.

M. Bachmayr
Sorbonne Universités, UPMC Univ Paris 06, CNRS, UMR 7598, Laboratoire Jacques-Louis Lions
4 place Jussieu, 75005, Paris, France
E-mail: bachmayr@ljll.math.upmc.fr

R. Schneider
Institut für Mathematik, Technische Universität Berlin
Straße des 17. Juni 136, 10623 Berlin, Germany
E-mail: schneidr@math.tu-berlin.de

A. Uschmajew
Hausdorff Center for Mathematics & Institute for Numerical Simulation
University of Bonn, 53115 Bonn, Germany
E-mail: uschmajew@ins.uni-bonn.de

strongly correlated regime, and to the computation of metastable states in molecular dynamics.

Keywords hierarchical tensors · low-rank approximation · high-dimensional partial differential equations

Mathematics Subject Classification (2000) 65-02 · 65F99 · 65J · 49M · 35C

1 Introduction

The numerical solution of high-dimensional partial differential equations remains one of the most challenging tasks in numerical mathematics. A naive discretisation based on well-established methods for solving PDEs numerically, such as finite differences, finite elements or spectral elements, suffers severely from the so-called *curse of dimensionality*. This notion refers to the exponential scaling $\mathcal{O}(n^d)$ of the computational complexity with respect to the dimension d of the discretised domain. For example, if $d = 10$ and we consider $n = 100$ basis functions in each coordinate direction, this leads to a discretisation space of dimension 100^{10}. Even for low-dimensional univariate spaces, e.g., $n = 2$, but with $d = 500$, one has to deal with a space of dimension 2^{500}. It is therefore clear that one needs to find additional structures to design tractable methods for such large-scale problems.

Many established methods for large-scale problems rely on the framework of sparse and nonlinear approximation theory in certain dictionaries [37]. These dictionaries are fixed in advance, and their appropriate choice is crucial. Low-rank approximation can be regarded as a related approach, but with the dictionary consisting of general separable functions – going back to one of the oldest ideas in applied mathematics, namely *separation of variables*. As this dictionary is uncountable large, the actual basis functions used for a given problem have to be computed adaptively.

On the level of matrix- or bivariate approximation, the *singular value decomposition* (SVD) provides a tool to find such problem-adapted, separable basis functions. Related concepts underlie model order reduction techniques such as *proper orthogonal decompositions* and *reduced bases* [115]. In fact, one might say that low-rank matrix approximation is one of the most versatile concepts in computational sciences. Generalizing these principles to higher-order tensor has proven to be a promising, yet nontrivial, way to tackle high-dimensional problems and multivariate functions [83, 16, 61]. This article presents a survey of low-rank tensor techniques from the perspective of hierarchical tensors, and complements former review articles [83, 63, 59, 66] with novel aspects. A more detailed review of tensor networks for signal processing and big data applications, with detailed explanations and visualizations for all prominent low-rank tensor formats can be found in [25]. For an exhaustive treatment, we also recommend the monograph [61].

Regarding low-rank decomposition, the transition from linear to multi-linear algebra, is not as straightforward and harmless as one might expect. The *canonical polyadic format* [72] represents a tensor $\mathbf{u}$ of order d as a sum of elementary tensor

products, or rank-one tensors,

$$\mathbf{u}(i_1,\dots,i_d) = \sum_{k=1}^{r} \mathbf{C}_k^1(i_1)\cdots\mathbf{C}_k^d(i_d), \quad i_\mu = 1,\dots,n_\mu,\ \mu = 1,\dots,d, \tag{1.1}$$

with $\mathbf{C}_k^\mu \in \mathbb{R}^{n_\mu}$. For tensors of order two, the CP format simply represents the factorization of a rank-r matrix, and therefore is a natural representation for higher-order tensors as well. Correspondingly, the minimal r required in (1.1) is called the *canonical rank* of $\mathbf{u}$.

If r is small, the CP representation (1.1) is extremely data-sparse. From the perspective of numerical analysis, however, it turns out to have several disadvantages in case $d > 2$. For example, the set of tensors of canonical at most r is not closed [123]. This is reflected by the fact that for most optimisation problems involving tensors of low CP rank no robust methods exist. For further results concerning difficulties with the CP representation and rank of higher-order tensors, we refer to [123, 71, 61, 132], and highlight the concise overview [96]. Many of these issues have also been investigated from the perspective of algebraic geometry, see the monograph [91].

The present article is intended to provide an introduction and a survey of a somehow alternative route. Instead of directly extending matrices techniques to analogous notions for tensors, the strategy here is to reduce questions of tensor approximation to matrix analysis. This can be accomplished by the *hierarchical tensor* (HT) format, introduced by Hackbusch and Kühn [65], and the *tensor train* (TT) format, developed by Oseledets and Tyrtyshnikov [106, 110, 109, 107]. They provide alternative data-sparse tensor decompositions with stability properties comparable to the SVD in the matrix case, and can be regarded as multi-level versions of the Tucker format [129, 83]. Whereas the data complexity of the Tucker format intrinsically suffers from an exponential scaling with respect to dimensionality, the HT and TT format have the potential of bringing this down to a linear scaling, as long as the ranks are moderate. This compromise between numerical stability and potential data sparsity makes the HT and TT formats promising model class for representing and approximating tensors.

However, circumventing the curse of dimensionality by introducing a non-linear (here: multi-linear) parameterization comes at the price of introducing a *curse of nonlinearity*, or more precisely, a *curse of non-convexity*. Our model class of low rank hierarchical tensors is no longer a linear space nor a convex set. Therefore, it becomes notoriously difficult to find globally optimal solutions to approximation problems, and first-order optimality conditions remain local. In principle, the explicit multi-linear representation of hierarchical tensors is amenable to block optimisation techniques like variants of the celebrated *alternating least squares* method, e.g. [23, 68, 35, 16, 31, 83, 74, 87, 128, 108, 142, 41], but their convergence analysis is typically a challenging task as the multilinear structure does not meet classical textbook assumptions on block optimisation. Another class of local optimisation algorithms can be designed using the fact that, at least for fixed rank parameters, the model class is a smooth embedded manifold in tensor space, and explicit descriptions of its tangent space are available [82, 75, 134, 101, 67, 4, 135, 85, 32, 100]. However, here one his faced with a technical difficulty that this manifold is not closed: its closure only constitutes an algebraic variety [119].

An important tool available for hierarchical tensor representation is the *hierarchical singular value decomposition* (HSVD) [56], as it can be used to find a quasi-best low-rank approximation using only matrix procedures with full error control. The HSVD is an extension of the *higher-order singular value decomposition* [36] to different types of hierarchical tensor models, including TT [109,105,61]. This enables the construction of iterative methods based on low-rank truncations of iterates, such as tensor variants of iterative singular value thresholding algorithms [81,64,86,10,8,17].

Historically, the parameterization in a hierarchical tensor framework has evolved independently in the quantum physics community, in the form of *renormalization group* ideas [138,51], and more explicitly in the framework of *matrix product* and *tensor network states* [120], including the HSVD for matrix product states [136]. An further independent source of such developments can also be found in quantum dynamics, with the *multi-layer multi-configurational time dependent Hartree* (MCTDH) method [14,137,98]. We refer the interested reader to the survey articles [59,95,126] and to the monograph [61].

Although the resulting tensor representations have been used in different contexts, the perspective of hierarchical subspace approximation in [65] and [61] seems to be completely new. Here, we would like to outline how this concept enables one to overcome most of the difficulties with the parameterization by the canonical format. Most of the important properties of hierarchical tensors can easily be deduced from the underlying very basic definitions. For a more detailed analysis, we refer to the respective original papers. We do not aim to give a complete treatment, rather but to demonstrate the potential of hierarchical low-rank tensor representations from their basic principles. They provide a universal and versatile tool, with basic algorithms that are relatively simple to implement (requiring only basic linear algebra operations) and easily adaptable to various different settings.

An application of hierarchical tensors of particular interest, on which we focus here, is the treatment of high-dimensional partial differential equations. In this article, we will consider three major examples in further detail: PDEs depending on countably many parameters, which arise in particular in deterministic formulations of stochastic problems; the many-particle Schrödinger equation in quantum physics; and the Fokker-Planck equation describing a mechanical system in a stochastic environment. A further example of an application of practical importance are *chemical master equations*, for which we refer to [38,39].

This article is arranged as follows: Section 2 covers basic notions of low-rank expansions and tensor networks. In Section 3 we consider subspace-based representations and basic properties of hierarchical tensor representations, which play a role in the algorithms using fixed hierarchical ranks discussed in Section 4. In Section 5, we turn to questions of convergence of hierarchical tensor approximations with respect to the ranks, and consider thresholding algorithms operating on representations of variable ranks in Section 6. Finally, in Section 7, we describe in more detail the mentioned applications to high-dimensional PDEs.

2 Tensor product parameterization

In this section, we consider basic notions of low-rank tensor formats and tensor networks and how linear algebra operations can be carried out on such representations.

2.1 Tensor product spaces and multivariate functions

We start with some preliminaries. In this paper, we consider the d-fold topological tensor product

$$\mathscr{V} = \bigotimes_{\mu=1}^{d} \mathscr{V}_\mu \tag{2.1}$$

of separable $\mathbb{K}$-Hilbert spaces $\mathscr{V}_1, \dots, \mathscr{V}_d$. For concreteness, we will focus on the real field $\mathbb{K} = \mathbb{R}$, although many parts are easy to extend to the complex field $\mathbb{K} = \mathbb{C}$. The confinement to Hilbert spaces constitutes a certain restriction, but still covers are broad range of applications. The topological difficulties that arise in a general Banach space setting are beyond the scope of the present paper, see [48, 61]. Avoiding them allows us to put clearer focus on the numerical aspect of tensor product approximation.

We do not give the definition the topological tensor product of Hilbert spaces (2.1) in full detail (see [61]), but only recall the properties necessary for our later purposes. Let $n_\mu \in \mathbb{N} \cup \{\infty\}$ be the dimension of $\mathscr{V}_\mu$. We set

$$\mathscr{I}_\mu = \begin{cases} \{1, \dots, n_\mu\} & , \quad \text{if } n_\mu < \infty, \\ \mathbb{N} & , \quad \text{else,} \end{cases}$$

and $\mathscr{I} = \mathscr{I}_1 \times \cdots \times \mathscr{I}_d$. Fixing an orthonormal basis $\{\mathbf{e}^\mu_{i_\mu} : i_\mu \in \mathscr{I}_\mu\}$ for each $\mathscr{V}_\mu$, we obtain a unitary isomorphism $\varphi^\mu \colon \ell^2(\mathscr{I}_\mu) \to \mathscr{V}_\mu$ by

$$\varphi^\mu(\mathbf{c}) := \sum_{i \in \mathscr{I}_\mu} \mathbf{c}(i)\, \mathbf{e}^\mu_i, \quad \mathbf{c} \in \ell^2(\mathscr{I}_\mu).$$

Then $\{\mathbf{e}^1_{i_1} \otimes \cdots \otimes \mathbf{e}^d_{i_d} : i_1, \dots, i_d \in \mathscr{I}\}$ is an orthonormal basis of $\mathscr{V}$, and $\Phi := \varphi^1 \otimes \cdots \otimes \varphi^d$ is a unitary isomorphism from $\ell^2(\mathscr{I})$ to $\mathscr{V}$.

Such a fixed choice of orthonormal basis allows us to identify the elements of $\mathscr{V}$ with their coefficient tensors $\mathbf{u} \in \ell^2(\mathscr{I})$,

$$(i_1, \dots, i_d) \mapsto \mathbf{u}(i_1, \dots, i_d) \in \mathbb{R} \quad i_\mu \in \mathscr{I}_\mu; \;\; \mu = 1, \dots, d,$$

often called *hypermatrices*, depending on *discrete variables* i_μ, usually called *indices*.

In conclusion, we will focus in the following on the space $\ell^2(\mathscr{I})$, which is itself a tensor product of Hilbert spaces, namely,

$$\ell^2(\mathscr{I}) = \ell^2(\mathscr{I}_1) \otimes \cdots \otimes \ell^2(\mathscr{I}_d). \tag{2.2}$$

Note that the corresponding multilinear tensor product map of d univariate ℓ^2-functions is defined pointwise as $(\mathbf{u}^1 \otimes \cdots \otimes \mathbf{u}^d)(i_1, \dots, i_d) = \mathbf{u}^1(i_1) \cdots \mathbf{u}^d(i_d)$. Tensors of this

form are called *elementary tensors* or *rank-one tensors*. Also the terminology *decomposable tensors* is used in differential geometry.

Let $n = \max\{n_\mu : \mu = 1,\dots,d\}$, be the maximal dimension of $\mathscr{V}_\mu$. Then the number of possibly non-zero entries in the representation of $\mathbf{u}$ is $n_1 \cdots n_d = \mathscr{O}(n^d)$. This is often referred to as the *curse of dimensionality*. In the present paper we intend to circumvent the curse of dimensionality. Indeed this is a common issue in the previously mentioned examples of high-dimensional PDEs.

In very abstract terms, all low-rank tensor decompositions considered below ultimately decompose the tensor $\mathbf{u} \in \ell^2(\mathscr{I})$ such that

$$\mathbf{u}(i_1,\dots,i_d) = \tau(\mathbf{C}^1(i_1),\dots,\mathbf{C}^d(i_d),\mathbf{C}^{d+1},\dots,\mathbf{C}^D), \tag{2.3}$$

where $\tau : \mathscr{W} := \mathscr{W}_1 \times \dots \mathscr{W}_d \times \mathscr{W}_{d+1} \times \cdots \times \mathscr{W}_D \to \mathbb{R}$ is *multilinear* on a Cartesian product of vector spaces $\mathscr{W}_\nu$, $\nu = 1,\dots,D$. The choice of these vector spaces and the map τ determines the format, and the tensors in its range are considered as "low-rank". An example is the CP representation (1.1).

Remark 2.1 Since Φ is multilinear as well, we obtain a representations of the very same multilinear structure (2.3) for the corresponding elements of $\mathscr{V}$,

$$\Phi(\mathbf{u}) = (\varphi^1 \otimes \cdots \otimes \varphi^d)\big((\xi_1,\dots,\xi_d) \mapsto \tau(\mathbf{C}^1(\xi_1),\dots,\mathbf{C}^d(\xi_d),\mathbf{C}^{d+1},\dots,\mathbf{C}^D)\big).$$

For instance, if $\mathscr{V}$ is a function space on a tensor product domain $D = D_1 \times \cdots \times D_d$ on which point evaluation is defined, and $(\mathbf{e}^1 \otimes \cdots \otimes \mathbf{e}^d)(x_1,\dots,x_d) = \mathbf{e}^1(x_1)\cdots\mathbf{e}^d(x_d)$ for $x \in D$, then *formally* (dispensing for the moment with possible convergence issues), exploiting the multilinearity properties, we obtain

$$\begin{aligned}\Phi(\mathbf{u})(x_1,\dots,x_d) &= \tau\left(\sum_{i_1=1}^{n_1} \mathbf{e}^1_{i_1}(x_1)\mathbf{C}^1(i_1),\dots,\sum_{i_d=1}^{n_d} \mathbf{e}^d_{i_d}(x_d)\mathbf{C}^d(i_d),\mathbf{C}^{d+1},\dots,\mathbf{C}^D\right)\\ &= \tau(\varphi^1(\mathbf{C}^1)(x_1),\dots,\varphi^d(\mathbf{C}^d)(x_d),\mathbf{C}^{d+1},\dots,\mathbf{C}^D),\end{aligned}$$

and the same applies to other tensor product functionals on $\mathscr{V}$. Since in the present case of Hilbert spaces (2.1), the identification with $\ell^2(\mathscr{I})$ via Φ thus also preserves the considered low-rank structures, we exclusively work on basis representations in $\ell^2(\mathscr{I})$ in what follows.

2.2 The canonical tensor format

The *canonical tensor format*, also called *CP (canonical polyadic) decomposition*, *CANDECOMP* or *PARAFAC*, represents a tensor of order d as a sum of elementary tensor products $\mathbf{u} = \sum_{k=1}^r \mathbf{c}^1_k \otimes \cdots \otimes \mathbf{c}^d_k$, that is

$$\mathbf{u}(i_1,\dots,i_d) = \sum_{k=1}^r \mathbf{C}^1(i_1,k)\cdots\mathbf{C}^d(i_d,k), \tag{2.4}$$

with $\mathbf{c}^\mu_k = \mathbf{C}^\mu(\cdot,k) \in \ell^2(\mathscr{I}_\mu)$ [72,23,68]. The minimal r such that such a decomposition exists is called the *canonical rank* (or simply *rank*) of the tensor $\mathbf{u}$. It can be infinite.

Depending on the rank, the representation in the canonical tensor format has a potentially extremely low complexity. Namely, it requires at most rdn nonzero entries, where $n = \max |\mathscr{I}_\mu|$. Another feature (in the case $d > 2$) are the strong uniqueness properties of the representation (assuming the r equals the rank) which led to its big success in signal processing and data analysis, see [89, 83, 26] and references therein.

In view of the present motivating high-dimensional partial differential equation, one can observe that the involved operator can usually be represented in the form of a canonical tensor operator, and the right hand side is also very often of the form above. This implies that the operator and the right hand sides can be stored within this data-sparse representation. This motivates the basic assumption in numerical tensor calculus that all input data can be represented in a sparse tensor form. Then there is a reasonable hope that the solution of such a high-dimensional PDE might also be approximated by a tensor in the canonical tensor format with moderate r. The precise justification for this hope is subject to ongoing research, but many known numerical solutions obtained by tensor product ansatz functions like (trigonometric) polynomials, sparse grids, Gaussian kernels and so on are in fact low-rank approximations, mostly in the canonical format. However, the key idea in non-linear low-rank approximation is to not fix possible basis functions in (2.4) in advance. Then we have an extremely large library of functions at our disposal. Motivated by the seminal papers [15, 16], we will pursue this idea throughout the present article.

From a theoretical viewpoint, the canonical tensor representation (2.4) is a straightforward generalisation of low rank matrix representation, as it coincides with it when $d = 2$. As it turns out, however, the parameterization of tensors via the canonical representation (2.4) is not as harmless as it seems to be. For example, for $d > 2$, the following difficulties appear:

- The canonical tensor rank is (in case of finite-dimensional spaces) NP-hard to compute [71].
- The set of tensors of the above form with canonical rank at most r is not closed [123] (border rank problem). As a consequence, a best approximation of a tensor by one of smaller canonical rank might not exist. This is in strong contrast to the matrix case $d = 2$, see Sec. 3.1.
- In fact, the set of tensor of rank at most r does not form an algebraic variety [91].

Further surprising and fascinating difficulties with the canonical tensor rank in case $d > 2$ with references are listed in [90, 83, 132, 96]. Deep theory of *algebraic geometry* has been invoked for the investigation of these problems, see the monograph [91] for the state of the art. The problem of non-closedness can be often cured by imposing further conditions such symmetry [91], nonnegativity [97] or norm bounds on factors [123].

In this paper we show a way to avoid all these difficulties by considering another type of low-rank representation, namely the *hierarchical tensor representation* [61], but at the price of a slightly higher computational and conceptual complexity. Roughly speaking, the principle of hierarchical tensor representations is to reduce the treatment of higher-order tensors to matrix analysis.

2.3 Low-rank formats via additional contractions

For fixed choice of r, the canonical tensor format is multilinear with respect to every matrix $\mathbf{C}^\mu := \big(\mathbf{C}^\mu(i,k)\big)_{i\in\mathscr{I}_\mu,k=1,\dots,r}$. A generalised concept of low-rank tensor formats consists in considering classes of tensors which are images of more general multilinear parameterization. They can be formally derived as extension of the canonical format, by allowing the $\mathbf{C}^\mu$ to potentially depend on more *contraction variables* $k_1,\dots,k_E$, that is, $\mathbf{C}^\mu(i_\mu,k_1,\dots,k_E)$. For clarity, we will call the indices i_μ *physical variables*. We may even introduce components $\mathbf{C}^\nu(k_1,\dots,k_E)$, $d+1\le\nu\le D$, that do not depend on any physical variable. By summing over all the newly introduced contraction indices over a fixed range $k_\eta = 1,\dots,r_\eta$, we obtain the tensor representation

$$\mathbf{u}(i_1,\dots,i_d) = \sum_{k_1=1}^{r_1}\cdots\sum_{k_m=1}^{r_E}\prod_{\mu=1}^{d}\mathbf{C}^\mu(i_\mu,k_1,\dots,k_E)\prod_{\nu=d+1}^{D}\mathbf{C}^\nu(k_1,\dots,k_m), \tag{2.5}$$

which separates the physical variables. Again, we can regard $\mathbf{u}$ as elements of the image of a multilinear map $\tau_\mathfrak{r}$,

$$\mathbf{u} = \tau_\mathfrak{r}(\mathbf{C}^1,\dots,\mathbf{C}^D), \tag{2.6}$$

parametrizing a certain class of "low-rank" tensors. By $\mathfrak{r} = (r_1,\dots,r_E)$ we indicate that this map τ, and with it the notion of rank, depends on the *representation ranks* $r_1,\dots,r_E$.

The disadvantage compared to the canonical format is that the component tensors have order p_ν instead of 2, where $1\le p_\nu\le D$ is the number of contraction variables in $\mathbf{C}^\nu$ which are actually active.[1] In cases of interest introduced below (like the HT or TT format), this number is small, say $p=\max\{p_\nu:\nu=1,\dots,D\}\le 3$, so that we do not sacrifice too much in terms of complexity. With $r=\max\{r_\eta:\eta=1,\dots,E\}$, the data complexity of the format (2.5) is bounded by $\mathscr{O}(nDr^p)$.

2.4 Tensor networks

Among the general low-rank formats (2.5) we will confine to a subclass where the contractions, that is, the summations over the contraction variables can be graphically visualized as a tensor network.

Tensor networks are a useful and versatile graphical tool for describing decompositions of multi-variate functions (tensors) into nested summations over contraction indices. In principle, they are easy to understand, but a rigorous description can be tedious. A possible way is consider the low-rank formats (2.5) that separate the physical indices as a special case of the following more general form

$$\mathbf{u}(i_1,\dots,i_d) = \sum_{k_1=1}^{r_1}\cdots\sum_{k_E=1}^{r_E}\prod_{\nu=1}^{D}\mathbf{C}^\nu(i_1,\dots,i_d,k_1,\dots,k_E). \tag{2.7}$$

[1] A contraction index k_η is called *inactive* in $\mathbf{C}^\nu$, if $\mathbf{C}^\nu$ does not depend on this index. The other indices are called *active*. The notation will be adjusted to reflect the dependence on active indices only later for special cases.

The main difference to (2.5) is that the components may depend on more than one physical index.

Definition 2.2 We call the multilinear parameterization (2.7) a *tensor network*, if

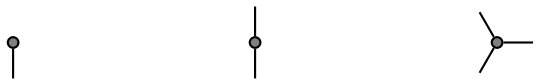
(i) each physical variable i_μ, $\mu = 1,\dots,d$, is active in exactly one component $\mathbf{C}^\nu$;
(ii) each contraction index k_η, $\eta = 1,\dots,E$, is active in precisely two components $\mathbf{C}^\nu$, $1 \le \nu \le D$,

see also [46, 120] and the references given there.

We note that the canonical tensor format (2.4) is *not* a tensor network, since the contraction variable k relates to all physical variables i_μ.

Tensor networks can be represented as graphs with nodes $\nu = 1,\dots,D$, representing component $\mathbf{C}^\nu$, and every contraction index k_η, $\eta = 1,\dots,E$, representing an edge connecting the two nodes in which they are active. In this way, edges connecting to nodes represent a summation over the corresponding contraction variable. Among all nodes, the ones in which a physical variable i_μ, $\mu = 1,\dots,d$, is active, play a special role and get an additional label, which in our pictures will be visualized by an additional open edge connected to the node. Conversely, the number of open edges in a tensor network determines the order of tensors under considerations.

The basic examples of tensor networks are plain vectors, matrices and higher-order tensors without any contraction indices. For example a vector $i \mapsto \mathbf{u}(i)$ is a node with single edge i, a matrix $(i_1,i_2) \mapsto \mathbf{U}(i_1,i_2)$ is a node with two edges i_1,i_2, and a d-th order tensor is a node with d edges connected to it:



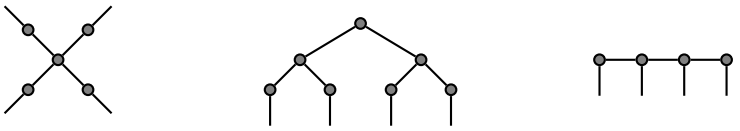
vector matrix third-order tensor

Low-rank matrix decompositions like $\mathbf{A} = \mathbf{U}\mathbf{V}^T$ or $\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^T$ are tensor networks, the latter containing a node with no physical index:

$\mathbf{U}\mathbf{V}^T$ $\mathbf{U}\Sigma\mathbf{V}^T$

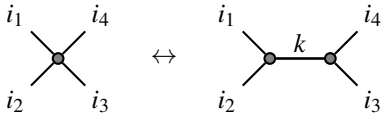
Note that these graphs to not show which physical variables belong to which open edge. To emphasize a concrete choice one can attach the labels n_μ to them. Also the range of the contraction indices can be specified. As an example, we illustrate how to contract and decontract a tensor of order $d = 4$ by a rank-r matrix decomposition that separates physical indices (i_1,i_2) from (i_3,i_4), using, e.g., SVD:

$$\mathbf{u}(i_1,i_2,i_3,i_4) = \sum_{k=1}^{r} \mathbf{C}^1(i_1,i_2,k)\mathbf{C}^2(i_3,i_4,k)$$



(a) Tucker format (b) hierarchical Tucker (HT) format (c) tensor train (TT) format

Fig. 2.1 Important examples of (tree) tensor networks.



Diagrammatic representations of a similar kind are also common in quantum physics for keeping track of summations, for instance Feynman and Goldstein diagrams.

It is easy to estimate the storage complexity of a tensor network. Every node requires the storage of a tensor whose order equals the number of edges connected to it. Let again p be a bound for the number of connected (active) contraction variables for a node, and q a bound for the number of physical indices. Let again $n_\mu \leq n$ for all μ, and $r_\eta \leq r$ for all η, then the storage requirement for every node is bounded by $n^q r^p$. Computationally efficient tensor network representations of multi-variate functions arise by bounding p and r, and only considering networks with $q = 1$, that is, with strict separation of all physical indices (in the diagram a node then has at most one open edge). Such tensor networks form a subset of the general "low-rank" formats considered in (2.5). The Tucker, hierarchical Tucker (HT), and tensor train (TT) formats are examples, and will be treated in detail in Sec. 3. In case $d = 4$, they are represented by the tensor networks depicted in Fig. 2.1.

By allowing large enough representation ranks, it is always possible to represent a d-th order tensor in any of these formats, but the required values $r = \max_{r_\eta}$ can differ substantially depending on the choice of format. A potential disadvantage of the Tucker format is that $p = d$, which implies a curse of dimension for large d. In contrast, $p = 3$ in HT and TT.

An important property of the Tucker, HT and TT format is that they are tree tensor networks.

Definition 2.3 A tensor network is called a *tree tensor network* if its graph is a tree, that is, contains no loops.

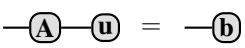
Among the general tensor networks, the networks with tree structure have favorable topological properties that make them more amenable to numerical utilization. For instance, tensors representable in tree networks with fixed rank bounds r_ν form closed sets, and the ranks have clear interpretation as matrix ranks, as will be explained in Section 3. In contrast, it has been shown that the set of tensors represented by a tensor network whose graph has closed loops is not closed in the Zariski sense [92]. In fact, there is no evidence that the more general tensor network parameterization with loops

do not suffer from similar problems as the canonical tensor format, which is not even a tensor network. In the following, we therefore restrict to tree tensor networks.

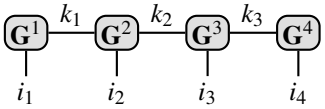
2.5 Linear algebra operations

For two tensors given in the same tree tensor network presentation it is easy to perform standard linear algebra operations, such as summation, Hadamard (pointwise) product, and inner products. Also the application of a linear operator to such a tensor can be performed in this representation if the operator is in the "compatible" form.

For instance, a matrix-vector product $\mathbf{b} = \mathbf{A}\mathbf{u}$ results in a vector, and is obtained as a single contraction $\mathbf{b}(i) = \sum_{k=1}^{n} \mathbf{A}(i,k)\mathbf{u}(k)$. Hence it has the following tensor network representation.

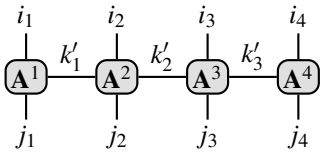


As a further illustration, consider a fourth-order tensor represented in the TT format (see Sec. 3.4):

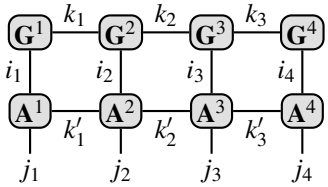


$$\mathbf{u}(i_1,i_2,i_3,i_4) = \sum_{k_1=1}^{r_1} \sum_{k_2=1}^{r_2} \sum_{k_3=1}^{r_3} \mathbf{G}^1(i_1,k_1)\mathbf{G}^2(k_1,i_2,k_2)\mathbf{G}^3(k_2,i_3,k_3)\mathbf{G}^4(k_3,i_4)$$

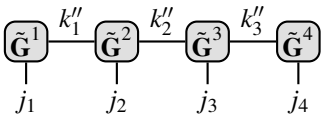
A linear operator $\mathbf{A}$ in TT format has the following form:



The application of $\mathbf{A}$ to $\mathbf{u}$ is illustrated by:



Summing over connected edges n_μ related to physical variables results again in a TT tensor $\tilde{\mathbf{u}} = \mathbf{A}\mathbf{u}$, but with possibly larger $R_\eta \le r_\eta s_\eta$ instead of r_η,



$$\tilde{\mathbf{u}}(j_1,j_2,j_3,j_4) = \sum_{k_1''=1}^{R_1}\sum_{k_2''=1}^{R_2}\sum_{k_3''=1}^{R_3} \tilde{\mathbf{G}}^1(j_1,k_1'')\tilde{\mathbf{G}}^2(k_1'',j_2,k_2'')\tilde{\mathbf{G}}^3(k_2'',j_3,k_3'')\tilde{\mathbf{G}}^4(k_3'',j_4).$$

It can be seen that the overall complexity of computing $\mathbf{Au}$ is linear in d, quadratic in n and only polynomial in the ranks.

To estimate the complexity of standard linear algebra operations, one observes that summing tensor in tree network representations leads to summation of ranks, while multiplicative operations like matrix-vector products or Hadamard products lead to multiplication of ranks. Luckily, this is only an upper estimate for the ranks. How to recompress the resulting parameterization with and without loss of accuracy will be shown later. Details about linear algebra operations are beyond the scope of this paper, but can be found in [61, 106, 25].

3 Tree tensor networks as nested subspace representations

In this section we explain how the deficiencies of the canonical format are cured using tree tensor network parameterizations. Tree tensor networks have the fundamental property that if one edge of the tree is removed, exactly two subtrees are obtained. This technique allows to apply matrix techniques to tree tensor networks and constitutes the main difference to the canonical tensor format.

3.1 The matrix case $d = 2$ revisited

An $m \times n$ can be either seen as an element of $\mathbb{R}^m \otimes \mathbb{R}^n$, a bivariate function, or as a linear operator from $\mathbb{R}^n$ to $\mathbb{R}^n$. In the general, possibly infinite-dimensional case this is generalised by the fact that the topological tensor product of $\mathscr{H} = \mathscr{H}_1 \otimes \mathscr{H}_2$ is isometrically isomorphic to the Hilbert space $\mathrm{HS}(\mathscr{H}_2, \mathscr{H}_1)$ of Hilbert-Schmidt operators from $\mathscr{H}_2$ to $\mathscr{H}_1$. This space consists of bounded linear operators $T : \mathscr{H}_2 \otimes \mathscr{H}_1$ for which $\|T\|_{\mathrm{HS}}^2 = \langle T, T\rangle_{\mathrm{HS}} < \infty$, where the inner product is defined as $\langle S, T\rangle_{\mathrm{HS}} = \sum_{i_2=1}^{n_2} \langle S\mathbf{e}_{i_2}^2, T\mathbf{e}_{i_2}^2\rangle$. Here $\{\mathbf{e}_{i_2}^\mu : i_2 \in \mathscr{I}_\mu\}$ is any orthonormal basis of $\mathscr{H}_2$. It is an easy exercise to convince oneself that the choice of basis is irrelevant. The isometric isomorphism $\mathbf{u} \mapsto T_{\mathbf{u}}$ between $\mathscr{H}$ and $\mathrm{HS}(\mathscr{H}_2, \mathscr{H}_1)$ we then consider is constructed by identifying

$$\mathbf{u}^1 \otimes \mathbf{u}^2 \in \mathscr{H}_1 \otimes \mathscr{H}_2 \quad \leftrightarrow \quad \langle \cdot, \mathbf{u}^2\rangle \mathbf{u}^1 \in \mathrm{HS}(\mathscr{H}_2, \mathscr{H}_1) \tag{3.1}$$

and linear expansion.

The relation to compact operators makes the case $d = 2$ unique as it enables spectral theory for obtaining tensor decompositions and low-rank approximations. The nuclear decomposition of compact operators plays the decisive role. It has been first obtained by Schmidt for integral operators [117]. A proof can be found in most textbooks on linear functional analysis or spectral theory. For matrices the decomposition (3.2) below is called the *singular value decomposition* (SVD), and can be traced back even further, see [124] for the history. We will use the same terminology. The best approximation property stated below was also obtained by Schmidt, and later also

attributed to Eckart and Young [42]. We state the result in $\ell^2(\mathscr{I}_1 \times \mathscr{I}_2)$; see [61] for a self-contained treatment from a more general tensor perspective.

Theorem 3.1 (E. Schmidt, 1907) *Let $\mathbf{u} \in \ell^2(\mathscr{I}_1 \times \mathscr{I}_2)$, then there exist orthonormal systems $\{\mathbf{U}^1(\cdot,k) : k \in \mathscr{I}_1\}$ in $\ell^2(\mathscr{I}_1)$ and $\{\mathbf{U}^2(\cdot,k) : k \in \mathscr{I}_2\}$ in $\ell^2(\mathscr{I}_2)$, and $\sigma_1 \geq \sigma_2 \geq \cdots \geq 0$, such that*

$$\mathbf{u}(i_1,i_2) = \sum_{k=1}^{\min(n_1,n_2)} \sigma_k \, \mathbf{U}^1(i_1,k)\mathbf{U}^2(i_2,k), \tag{3.2}$$

with convergence in $\ell^2(\mathscr{I}_1 \times \mathscr{I}_2)$. A best approximation of $\mathbf{u}$ by a tensor of rank $r \leq \min(n_1,n_2)$ in the norm of $\ell^2(\mathscr{I}_1 \times \mathscr{I}_2)$ is provided by

$$\mathbf{u}_r(i_1,i_2) = \sum_{k=1}^{r} \sigma_k \, \mathbf{U}^1(i_1,k)\mathbf{U}^2(i_2,k),$$

and approximation error satisfies

$$\|\mathbf{u} - \mathbf{u}_r\|^2 = \sum_{k=r+1}^{\min(n_1,n_2)} \sigma_k^2.$$

The best approximation is unique in case $\sigma_r > \sigma_{r+1}$.

The numbers σ_k are called *singular values*, the basis elements $\mathbf{U}^1(\cdot,k)$ and $\mathbf{U}^2(\cdot,k)$ are called corresponding left and right *singular vectors*. They are the eigenvectors of $T_{\mathbf{u}}T_{\mathbf{u}}^*$ and $T_{\mathbf{u}}^*T_{\mathbf{u}}$, respectively.

In matrix notation, let $\mathbf{U}$ be the (possibly infinite) matrix with entries $\mathbf{u}(i_1,i_2)$. Then, using (3.1), the singular value decomposition (3.2) takes the familiar form

$$\mathbf{U} = \mathbf{U}_1 \Sigma \mathbf{U}_2^T,$$

where $\mathbf{U}_\mu = [\mathbf{u}_1^\mu, \mathbf{u}_2^\mu, \dots]$ have columns $\mathbf{u}_k^\mu$, $\mu = 1,2$, and $\Sigma = \operatorname{diag}(\sigma_1, \sigma_2, \dots)$.

3.2 Subspace approximation

The problem of finding the best rank-r approximation to a tensor of order two (matrix) can be interpreted as a subspace approximation problem, and Schmidt's theorem 3.1 provides a solution.

The problem is as follows: Find subspaces $\mathscr{U}_1 \subseteq \ell^2(\mathscr{I}_1)$ and $\mathscr{U}_2 \subseteq \ell^2(\mathscr{I}_2)$ of dimension r such that

$$\operatorname{dist}(\mathbf{u}, \mathscr{U}_1 \otimes \mathscr{U}_2) = \|\mathbf{u} - \Pi_{\mathscr{U}_1 \otimes \mathscr{U}_2}\mathbf{u}\| = \min! \tag{3.3}$$

Here $\Pi_{\mathscr{U}_1 \otimes \mathscr{U}_2}$ denotes the orthogonal projection on $\mathscr{U}_1 \otimes \mathscr{U}_2$. The truncated singular value decomposition $\mathbf{u}_r$ is the solution to this problem, more precisely the subspaces spanned by the dominating r left and right singular vectors, respectively, since it holds that a tensor of order two is of at most r if and only it is contained in such a subspace

$\mathscr{U}_1 \otimes \mathscr{U}_2$.[2] We highlight that the admissible set over which we minimise the distance $\operatorname{dist}(\mathbf{u}, \mathscr{U}_1 \otimes \mathscr{U}_2)$ is the closure of a Cartesian product of Graßmannians [1,43,91]. Note that the rank of $\mathbf{u}$ now can be defined as the minimal r such that the minimal distance in (3.3) is zero.

In contrast to the representability in canonical tensor format, the interpretation of low-rank approximation as subspace approximation, which is possible in case $d = 2$, provides a different concept which offers advantageous mathematical properties also in the higher-order case. In the sequel we will pursue on this concept. A direct generalisation of (3.3) to higher-order tensors leads to the by now classical *Tucker format* [129,83,61]. Given a tensor $\mathbf{u} \in \ell^2(\mathscr{I})$ and dimensions $r_1, \dots, r_d$ one is searching for optimal subspaces $\mathscr{U}_\mu \subseteq \ell^2(\mathscr{I}_\mu)$ of dimension r_μ such that

$$\operatorname{dist}(\mathbf{u}, \mathscr{U}_1 \otimes \cdots \otimes \mathscr{U}_d) = \|\mathbf{u} - \Pi_{\mathscr{U}_1 \otimes \cdots \otimes \mathscr{U}_d} \mathbf{u}\| = \min! \tag{3.4}$$

The (elementwise) minimal tuple of $(r_1, \dots, r_d)$ which minimises the distance to zero is called the *Tucker rank* of $\mathbf{u}$. It follows from this definition that a tensor has Tucker rank at most $(r_1, \dots, r_d)$ if and only if $\mathbf{u} \in \mathscr{U}_1 \otimes \cdots \otimes \mathscr{U}_d$ with $\dim(\mathscr{U}_\mu) \leq r$. Note that this in turn is the case if and only if $\mathbf{u}$ can be written as

$$\mathbf{u}(i_1, \dots, i_d) = \sum_{k_1=1}^{r_1} \cdots \sum_{k_d=1}^{r_d} \mathbf{C}(i_1, \dots, i_d, k_1, \dots, k_d) \mathbf{U}^1(i_1, k_1) \cdots \mathbf{U}^d(i_d, k_d). \tag{3.5}$$

For instance, one can choose $\{\mathbf{U}^\mu(\cdot, 1), \dots, \mathbf{U}^d(\cdot, r)\}$ to be a basis of $\mathscr{U}_\mu$. The multilinear representation (3.5) of tensors is called the *Tucker format* [73,129]. Its tensor network representation is given in Fig. 2.1(a).

The minimal r_μ appearing in the Tucker rank of $\mathbf{u}$, as well as the corresponding subspaces $\mathscr{U}_\mu$ can be found constructively and independently from each other as follows. For $\mu = 1, \dots, d$, let $\mathscr{I}_\mu^c = \mathscr{I}_1 \times \cdots \times \mathscr{I}_{\mu-1} \times \mathscr{I}_{\mu+1} \times \cdots \times \mathscr{I}_d$. Then the spaces $\ell^2(\mathscr{I}_\mu \times \mathscr{I}_\mu^c) = \ell^2(\mathscr{I}_\mu) \otimes \ell^2(\mathscr{I}_\mu^c)$, which are tensor product spaces of order two, are all isometrically isomorphic to $\ell^2(\mathscr{I})$. The corresponding isomorphisms $\mathbf{u} \mapsto \mathbf{M}_\mu(\mathbf{u})$ are called matricisations. The SVDs of $\mathbf{M}_\mu(\mathbf{u})$ provide us with subspaces $\mathscr{U}_\mu$ of minimal dimension r_μ such that $\mathbf{M}_\mu(\mathbf{u}) \in \mathscr{U}_\mu \otimes \ell^2(\mathscr{I}_\mu^c)$, that is,

$$\mathbf{u} \in \ell^2(\mathscr{I}_1) \otimes \cdots \otimes \ell^2(\mathscr{I}_{\mu-1}) \otimes \mathscr{U}_\mu \otimes \ell^2(\mathscr{I}_{\mu+1}) \otimes \cdots \otimes \ell^2(\mathscr{I}_d). \tag{3.6}$$

Comparing with (3.4), this shows that this r_μ cannot be larger than the corresponding Tucker rank. On the other hand, since (3.6) holds for $\mu = 1, \dots, d$ simultaneously, we get (see, e.g., [61, Lemma 6.28])

$$\mathbf{u} \in \mathscr{U}_1 \otimes \cdots \otimes \mathscr{U}_d, \tag{3.7}$$

which in combination yields that the $\mathscr{U}_\mu$ found in this way solve (3.4). These considerations will be generalised to general tree tensor networks in Sec. 3.5.

Similar to the matrix case one may pose the problem of finding the best approximation of a tensor $\mathbf{u}$ by one of lower Tucker rank. This problem always has a solution [130,

[2] If $\mathbf{u} = \sum_{k=1}^r \mathbf{u}_k^1 \otimes \mathbf{u}_k^2$, then $\mathbf{u} \in \operatorname{span}\{\mathbf{u}_1^1, \dots, \mathbf{u}_r^1\} \otimes \operatorname{span}\{\mathbf{u}_1^2, \dots, \mathbf{u}_r^2\}$. Conversely, if $\mathbf{u}$ is in such a subspace, then there exist a_{ij} such that $\mathbf{u} = \sum_{i=1}^r \sum_{j=1}^r a_{ij} \mathbf{u}_i^1 \otimes \mathbf{u}_j^2 = \sum_{i=1}^r \mathbf{u}_i^1 \otimes \left(\sum_{j=1}^r a_{ij} \mathbf{u}_j^2 \right)$.

48], but no normal form providing a solution similar to the SVD is currently available. The higher-order SVD [36] uses the dominant left singular subspaces from the SVDs of the matricisations $\mathbf{M}_\mu(\mathbf{u})$, but they only provide quasi-optimal approximations. This will be explained in more detail Sec. 3.5, albeit somewhat more abstractly than probably necessary for the Tucker format.

There is a major drawback of the Tucker format, which motivates us to go beyond it: unless the *core tensor* $\mathbf{C}$ in (3.5) is sparse, the low rank Tucker representation does not prevent us from the curse of dimensionality. Since, in general, the core tensor contains $r_1 \cdots r_d$ possibly nonzero entries, its storage complexity scales exponentially with the dimensions as $\mathcal{O}(r^d)$ where $r = \max\{r_\mu : \mu = 1,\dots,d\}$. With $n = \max\{n_\mu : \mu = 1,\dots,d\}$, the overall complexity for storing the required data (including the basis vectors $\mathbf{U}^\mu(\cdot,k)$ is bounded by $\mathcal{O}(ndr + r^d)$. Without further sparsity of the core tensor, the pure Tucker format is appropriate for tensors of low order only, say $d \le 4$. Nonetheless, subspace based tensor representation approximation as in the Tucker format is not a dead end road. We will use it in a hierarchical fashion to circumvent the curse of dimensionality, at least for a large class of tensors.

3.3 Hierarchical tensor representation

The *hierarchical Tucker format* or *hierarchical tensor format* (HT) was introduced by Hackbusch and Kühn [65], and extends the idea of subspace approximation to a hierarchical or multi-level framework. It is a tree tensor network corresponding to the diagram in Fig. 2.1(b). Here we derive it from a geometric viewpoint. Let us

Reconsider the subspace relation (3.7) with subspaces $\mathcal{U}_\mu$ of dimension r_μ. For the representation of $\mathbf{u}$ it might be sufficient to have a $\mathcal{U}_{\{1,2\}} \subseteq \mathcal{U}_1 \otimes \mathcal{U}_2$ of possibly lower dimension $r_{\{1,2\}} \le r_1 r_2$ to ensure $\mathbf{u} \in \mathcal{U}_{\{1,2\}} \otimes \mathcal{U}_3 \otimes \cdots \otimes \mathcal{U}_d$. Then $\mathcal{U}_{\{1,2\}}$ is a space of "matrices", and has a basis $\{\mathbf{U}^{\{1,2\}}(\cdot,\cdot,k_{\{1,2\}}) : k_{\{1,2\}} = 1,\dots,r_{\{1,2\}}\}$, whose elements can be represented in the basis of $\mathcal{U}_1 \otimes \mathcal{U}_2$:

$$\mathbf{U}^{\{1,2\}}(i_1,i_2,k_{\{1,2\}}) = \sum_{k_1=1}^{r_1}\sum_{k_2=1}^{r_2} \mathbf{B}^{\{1,2\}}(k_1,k_2,k_{\{1,2\}})\mathbf{U}^1(i_1,k_1)\mathbf{U}^2(i_2,k_2).$$

One can now continue in several ways, e.g., by choosing a subspace $\mathcal{U}_{\{1,2,3\}} \subseteq \mathcal{U}_{\{1,2\}} \otimes \mathcal{U}_3 \subseteq \mathcal{U}_1 \otimes \mathcal{U}_2 \otimes \mathcal{U}_3$. another option is to find a subspace $\mathcal{U}_{\{1,2,3,4\}} \subseteq \mathcal{U}_{\{1,2\}} \otimes \mathcal{U}_{3,4}$, where $\mathcal{U}_{\{3,4\}}$ is defined analogously to $\mathcal{U}_{\{1,2\}}$, and so on.

For a systematic treatment, this approach is cast into the framework of a partition tree $\mathbb{T}$ (also called *dimension tree*) containing subsets of $\{1,\dots,d\}$ such that

(i) $\alpha^* := \{1,\dots,d\} \in \mathbb{T}$, and
(ii) for every $\alpha \in \mathbb{T}$ with $|\alpha| > 1$ there exist $\alpha_1, \alpha_2 \in \mathbb{T}$ such that $\alpha = \alpha_1 \cup \alpha_2$ and $\alpha_1 \cap \alpha_2 = \emptyset$.

Such a set $\mathbb{T}$ forms a binary tree by introducing edges between fathers and sons. The vertex α^* is then the root of this tree, while the singletons $\{\mu\}$, $\mu = 1,\dots,d$ are the leaves. By agreeing that α_1 should be the left son of α and α_2 the right son, a pre-order traversion through the tree yields the leaves $\{\mu\}$ appear according to a certain permutation $\Pi_\mathbb{T}$ of $\{1,\dots,d\}$.

By $\hat{\mathbb{T}}$ we denote the subset of α which are neither the root, nor a leaf (inner vertices). In the HT format, to every $\alpha \in \mathbb{T} \setminus \{\alpha^*\}$ with sons α_1, α_2 a subspace $\mathscr{U}_\alpha \subseteq \bigotimes_{j\in\alpha} \ell^2(\mathscr{I}_j)$ of dimension r_α is attached such that the *nestedness properties*

$$\mathscr{U}_\alpha \subseteq \mathscr{U}_{\alpha_1} \otimes \mathscr{U}_{\alpha_2}, \quad \alpha \in \hat{\mathbb{T}},$$

and $\pi_{\mathbb{T}}(\mathbf{u}) \in \mathscr{U}_{\alpha_1^*} \otimes \mathscr{U}_{\alpha_2^*}$ hold true. Here $\pi_{\mathbb{T}}$ denotes the natural isomorphism[3] between $\bigotimes_{\mu=1}^d \ell^2(\mathscr{I}_\mu)$ and $\bigotimes_{\mu=1}^d \ell^2(\mathscr{I}_{\Pi_{\mathbb{T}}(\mu)})$.

Corresponding bases $\{\mathbf{U}^\alpha(\cdot,\dots,\cdot,k_\alpha) : k_\alpha = 1,\dots,r_\alpha\}$ of $\mathscr{U}_\alpha$ are then recursively expressed as

$$\mathbf{U}^\alpha(\mathbf{i}_\alpha, k_\alpha) = \sum_{k_1=1}^{r_{\alpha_1}} \sum_{k_2=1}^{r_{\alpha_2}} \mathbf{B}^\alpha(k_1,k_2,k_\alpha)\mathbf{U}^{\alpha_1}(\mathbf{i}_{\alpha_1},k_1)\mathbf{U}^{\alpha_2}(\mathbf{i}_{\alpha_2},k_2), \quad \alpha \in \hat{\mathbb{T}}, \tag{3.8}$$

where $\mathbf{i}_\alpha = \times_{\mu\in\alpha}\{i_\mu\}$ denotes, with a slight abuse of notation, the tuple of physical variables represented by α. Finally, $\mathbf{u}$ is recovered as

$$\mathbf{u}(i_1,\dots,i_d) = \sum_{k_1=1}^{r_{\alpha_1^*}} \sum_{k_2=1}^{r_{\alpha_2^*}} \mathbf{B}^{\alpha^*}(k_1,k_2)\mathbf{U}^{\alpha_1^*}(\mathbf{i}_{\alpha_1^*},k_1)\mathbf{U}^{\alpha_2^*}(\mathbf{i}_{\alpha_2^*},k_2). \tag{3.9}$$

It will be notationally convenient to set $\mathbf{B}^\alpha = \mathbf{U}^\alpha$ for leaves $\alpha = \{\mu\}$. If equation (3.9) is recursively expanded using (3.8), we obtain a multilinear low-rank format of the form (2.5) with $E = |T| - 1$, $D = |T|$, $r_\eta = r_\alpha$, and $\mathbf{C}^\nu = \mathbf{B}^\alpha$ (in some ordering). Its network representation takes the form of the tree in Fig. 2.1(b), and has the same topology as the tree $\mathbb{T}$ itself, ignoring the edges with open ends which can be seen as labels indicating physical variables.

The tensors $\mathbf{B}^\alpha$ will be called *component tensors*, the terminology *transfer tensors* is also common in the literature. In line with (2.6) the tensors which are representable in the HT format with fixed $\mathfrak{r} = (r_\alpha)$ are the images

$$\mathbf{u} = \pi_{\mathbb{T}}^{-1}(\tau_{\mathbb{T},\mathfrak{r}}((\mathbf{B}^\alpha)_{\alpha\in\mathbb{T}}))$$

of a multilinear map $\pi_{\mathbb{T}}^{-1} \circ \tau_{\mathbb{T},\mathfrak{r}}$.

For fixed $\mathbf{u}$ and $\mathbb{T}$, the minimal possible r_α to represent $\mathbf{u}$ as image of $\tau_{\mathbb{T},\mathfrak{r}}$ are, as for the Tucker format, given by ranks of certain matricisations of $\mathbf{u}$. This will be explained in Sec. 3.5 for general tree tensor networks.

Depending on the contraction lengths r_α, the HT format can be efficient, as it only requires storing the tuple $(\mathbf{B}^\alpha)_{\alpha\in\mathbb{T}}$. Every $\mathbf{B}^\alpha$ is a tensor of order at most three. The number of nodes in the tree $\mathbb{T}$ is bounded by $2d-1 = \mathscr{O}(d)$, including the root node. Therefore the data complexity for representing $\mathbf{u}$ is $\mathscr{O}(ndr + dr^3)$, where $n = \max\{n_\mu : \mu = 1,\dots,d\}$, $r = \max\{r_\alpha : \alpha \in \mathbb{T}\setminus\{\alpha^*\}\}$. In contrast to the classical Tucker format it is formally no longer scaling exponentially with the order d.

It is straightforward to extend the concept to partition trees such that vertices are allowed to have more than two son, but the binary trees are the most common. Note that the Tucker format itself represents an extreme case where the root decomposes immediately into d leaves, see Fig. 2.1(a) again.

[3] One can think of $\pi_{\mathbb{T}}(\mathbf{u})$ as a reshape of the tensor $\mathbf{u}$ which relabels the physical variables according to the permutation Π induced by the order of the tree vertices. Note that in the pointwise formula (3.9) it is not needed.

3.4 Tensor trains and matrix product representation

As a third example, we consider the *tensor train* (TT) format as introduced in [107, 109]. As it later turned out, this format plays an important role in physics, where it is known as *matrix product states* (MPS). The unlabeled tree tensor network of this format can be seen in Fig. 2.1(c). When attaching the physical variable n_μ in natural order from left to right, the pointwise multilinear representation is

$$\mathbf{u}(i_1,\dots,i_d) = \sum_{k_1=1}^{r_1} \dots \sum_{k_{d-1}=1}^{r_{d-1}} \mathbf{G}^1(i_1,k_1)\mathbf{G}^2(k_1,i_2,k_2)\cdots\mathbf{G}^d(k_{d-1},i_d) \tag{3.10}$$

The TT format is hence of the form (2.5) with $D=d$ and $E=d-1$.

Introducing the matrices $\mathbf{G}^\mu(i_\mu) = [\mathbf{G}^\mu(k_{\mu-1},i_\mu,k_\mu)] \in \mathbb{R}^{r_{\mu-1}\times r_\mu}$, with the convention $r_0 = r_d = 1$, $\mathbf{G}^1(1,i_1,k_1) = \mathbf{G}^1(i_1,k_1)$, and $\mathbf{G}^d(k_{d-1},i_1,1) = \mathbf{G}^d(k_{d-1},i_d)$, formula (3.10) becomes a matrix product,

$$\mathbf{u}(i_1,\dots,i_d) = \mathbf{G}^1(i_1)\mathbf{G}^2(i_2)\cdots\mathbf{G}^d(i_d), \tag{3.11}$$

which explains the name matrix product states used in physics. In particular, the multilinear dependence on the components $\mathbf{G}^\mu$ is evident, and may be expressed as $\mathbf{u} = \tau_{\mathrm{TT}}(\mathbf{G}^1,\dots,\mathbf{G}^d)$.

From the viewpoint of subspace representation, the minimal r_η, $\eta = 1,\dots,d-1$, required for representing $\mathbf{u}$ in the TT format are the minimal dimensions of subspaces $\mathscr{U}_{\{1,\dots,\nu\}} \subseteq \bigotimes_{\mu=1}^{\nu} \ell^2(\mathscr{I}_\mu)$ such that the relations

$$\mathbf{u} \in \mathscr{U}_{\{1,\dots,\eta\}} \otimes \Big(\bigotimes_{\mu=\eta+1}^{d} \ell^2(\mathscr{I}_\mu)\Big), \quad \eta = 1,\dots,d-1$$

hold simultaneously. Again, these subspaces can be obtained as ranges of corresponding matricisations, as will be explained in the next subsection. Regarding nestedness, we will see that one even has

$$\mathscr{U}_{\{1,\dots,\eta\}} \subseteq \mathscr{U}_{\{1,\dots,\eta-1\}} \otimes \ell^2(\mathscr{I}_\eta), \quad \eta = 1,\dots,d-1.$$

A tensor in canonical format

$$\mathbf{u}(i_1,\dots,i_d) = \sum_{k=1}^{r_c} \mathbf{C}^1(i_1,k)\cdots\mathbf{C}^d(i_d,k)$$

can be easily written in TT form, by setting all r_η to r_c, $\mathbf{G}^1 = \mathbf{C}^1$, $\mathbf{C}^d = (\mathbf{G}^d)^T$, and

$$\mathbf{G}^\mu(k_{\mu-1},i_\mu,k_\mu) = \begin{cases} \mathbf{C}^\mu(i_\mu,k), & \text{if } k_{\mu-1} = k_\mu = k, \\ 0, & \text{else,} \end{cases}$$

for $\mu = 2,\dots,d-1$.. From (3.11) we conclude immediately that a single point evaluation $\mathbf{u}(i_1,\dots,i_d)$ can be computed easily by matrix multiplication using $\mathscr{O}(dr^2)$ arithmetic operations, where $r = \max\{r_\eta : \eta = 1,\dots,d-1\}$. With $n = \max\{n_\mu : \mu =$

$1,\dots,d\}$, the data required for the TT representation is $\mathcal{O}(dnr^2)$, as the d component tensors $\mathbf{G}^\mu$ need to be stored. Depending on r, the TT format hence offers the possibility to circumvent the curse of dimensionality.

Due to its convenient explicit representation (3.11) we will use the TT format frequently as a model case for explanation.

3.5 Matricisations and tree rank

After having discussed the most prominent examples of tree tensor networks in the previous sections, we return to the consideration of a general tree tensor network $\tau = \tau_{\mathfrak{r}}(\mathbf{C}^1,\dots,\mathbf{C}^D)$ encoding a representation (2.7) and obeying Definitions 2.2 and 2.3. The nodes have indices $1,\dots,D$, and the distribution of physical variables i_μ is fixed (it is allowed that nodes carry more than one physical index).

The topology of the network is described by the set of its edges. Following [9], we now introduce a notion of *effective edges*, which may in fact comprise several lines in a graphical representation such as Figure 2.1, and correspond precisely to the matricisations arising in the tensor format. The set of such edges will be denoted by $\mathbb{E}$. In slight deviation from (2.7), the contraction indices $(k_\eta)_{\eta\in\mathbb{E}}$ and the representation ranks $\mathfrak{r} = (r_\eta)_{\eta\in\mathbb{E}}$ will now be indexed by the set $\mathbb{E}$.

Since we are dealing with a tree tensor network, along every contraction index we may split the tree into two disjoint subtrees. Both subtrees must contain vertices carrying physical variables. Hence such a splitting induces a partition

$$\alpha^* = \{1,\dots,d\} = \alpha \cup \alpha^c$$

by gathering the μ for which the physical index i_μ is in the respective subtree. We then call the unordered pair $\{\alpha,\alpha^c\}$ an edge.

For instance, for a given partition tree $\mathbb{T}$ in the HT case, we have

$$\mathbb{E} = \big\{\{\alpha,\alpha^c\}\colon \alpha\in\mathbb{T}\setminus\{\alpha^*\}\big\} \tag{3.12}$$

as used in [9], with each element of $\mathbb{E}$ corresponding to precisely one matricisation arising in the format. As a consequence of the definition, for each $\eta\in\mathbb{E}$ we may pick a *representative* $[\eta]\in\mathbb{T}$. Note that in (3.12), the set $\{\alpha_1^*,\alpha_2^*\}$ appears twice as α runs over $\mathbb{T}\setminus\{\alpha^*\}$, which is a consequence of the two children of α^* corresponding to the same matricisation; hence $|\mathbb{E}| = 2d-3$.

In order to introduce the same notion for tree tensor networks, we first give a construction of a corresponding *generalised partition tree* $\mathbb{T}$ by assigning labels to the nodes in the tensor network as follows. Pick any node ν^* to be the root of the tree, for which we add $\alpha^* = \{1,\dots,d\}$ to $\mathbb{T}$. This induces a top-down (father-son) ordering in the whole tree. For all nodes ν, we have a partition of the physical variables in the respective subtree of the form

$$\alpha_\nu = \Big(\bigcup_{\nu'\in\mathrm{sons}(\nu)} \alpha_{\nu'}\Big)\cup\beta_\nu, \tag{3.13}$$

where β_ν is the set of physical variables attached to ν (of course allowing $\beta_\nu = \emptyset$). We now add all α_ν that are obtained recursively in this manner to the set $\mathbb{T}$. It is easy to see that such a labeling is possible for any choice of ν^*.

For such a generalised partition tree $\mathbb{T}$ of a tree tensor network, we again obtain a set of effective edges $\mathbb{E}$ exactly as in (3.12), and again have a representative $[\eta] \in \mathbb{T}$ for each $\eta \in \mathbb{E}$.

The difference of the general construction to the particular case (3.12) of the HT format is that we allow incomplete partitions (complemented by β_ν), and in principle also further nodes with the same label. In the case of the TT format (3.10), which corresponds to the network considered in Fig. 2.1(c) with linearly arranged i_μ, starting from the rightmost node $\nu^* = \{d\}$, one obtains the $d-1$ edges

$$\big\{\{1\},\{2,\dots,d\}\big\},\big\{\{1,2\},\{3,\dots,d\}\big\},\dots,\big\{\{1,\dots,d-1\},\{d\}\big\}$$

which in this case comprise the set $\mathbb{E}$.

The main purpose of this section is to show how the minimal representation ranks $(r_\eta)_{\eta\in\mathbb{E}}$ are obtained from matrix ranks. For every edge $\eta \in \mathbb{E}$, we have index sets $\mathscr{I}_\eta = \bigtimes_{\mu\in[\eta]} \mathscr{I}_\mu$ and $\mathscr{I}_\eta^c = \bigtimes_{\mu\in[\eta]^c} \mathscr{I}_\mu$, and, by (2.2), induces a natural isometric isomorphism

$$\mathbf{M}_\eta : \ell^2(\mathscr{I}) \to \ell^2(\mathscr{I}_\eta) \otimes \ell^2(\mathscr{I}_\eta^c),$$

called η-*matricisation* or simply *matricisation*. The second-order tensor $\mathbf{M}_\eta(\mathbf{u})$ represents a reshape of the hyper-matrix (array) $\mathbf{u}$ into a matrix in which the rows are indexed by $\mathscr{I}_\eta$ and the columns by $\mathscr{I}_\eta^c$. The order in which these index sets are traversed is unimportant for what follows.

Definition 3.2 The rank of $\mathbf{M}_\eta(\mathbf{u})$ is called the η-*rank* of $\mathbf{u}$, and denoted by $\operatorname{rank}_\eta(\mathbf{u})$. The tuple $\operatorname{rank}_\mathbb{E}(\mathbf{u}) = (\operatorname{rank}_\eta(\mathbf{u}))_{\eta\in\mathbb{E}}$ is called the *tree rank* of $\mathbf{u}$ for the given tree tensor network.

Theorem 3.3 *A tensor is representable in a tree tensor network $\tau_\mathfrak{t}$ with edges $\mathbb{E}$ if and only if* $\operatorname{rank}_\eta(\mathbf{u}) \le r_\eta$ *for all* $\eta \in \mathbb{E}$.

Proof Assume $\mathbf{u}$ is representable in the form (2.7). Extracting edge η corresponding to (without loss of generality, only one) contraction index k_η from the tree we obtain two disjoint subtrees on both sides of η, with corresponding contraction indices relabelled as $k_1,\dots,k_s$ and $k_{s+1},\dots,k_{E-1}$, respectively; the set of nodes for the components is partitioned into $\{1,\dots,D\} = \gamma'_\eta \cup \gamma''_\eta$. Since in every component $\mathbf{C}^\nu$ at most two contraction indices are active, it follows that

$$\mathbf{u}(i_1,\dots,i_d) = \sum_{k_\eta=1}^{r_\eta} \Big(\sum_{k_1=1}^{r_1} \cdots \sum_{k_s=1}^{r_s} \prod_{\nu'\in\gamma'} \mathbf{C}^{\nu'}(i_1,\dots,i_d,k_1,\dots,k_E) \Big) \times \Big(\sum_{k_{s+1}}^{r_{s+1}} \cdots \sum_{k_{E-1}=1}^{r_{E-1}} \prod_{\nu''\in\gamma''} \mathbf{C}^{\nu''}(i_1,\dots,i_d,k_1,\dots,k_E) \Big). \quad (3.14)$$

The edge η is of the form $\eta = \{\alpha,\alpha^c\}$, where all physical indices i_μ with $\mu \in \alpha$ are active in some $\mathbf{C}^{\nu'}$ with $\nu' \in \gamma'$, and those in α^c are active in some $\mathbf{C}^{\nu''}$ with $\nu'' \in \gamma''$. Thus (3.14) implies $\operatorname{rank}_\eta(\mathbf{u}) \le r_\eta$.

To prove the converse statement it suffices to show that we can choose $r_\eta = \operatorname{rank}_\eta(\mathbf{u})$. We assume a proper labelling with distinguished node ν^*. To every edge η belongs a subspace $\mathscr{U}_\eta \subseteq \ell^2(\mathscr{I}_{[\eta]})$, which is the Hilbert space whose orthonormal basis are the left singular vectors of $\mathbf{M}_\eta(\mathbf{u})$ belonging to positive singular values. Its dimension is r_η. In a slight abuse of notation (one has to involve an isomorphism correcting the permutation of factors in the tensor product) we note that

$$\mathbf{u} \in \mathscr{U}_\eta \otimes \ell^2(\mathscr{I}_\eta^c) \tag{3.15}$$

for every η. Here our argumentation will be rather informal to avoid notational technicalities. One can show that (3.15) in combination with (3.13) yields (in intuitive notation)

$$\mathscr{U}_{\alpha_\nu} \subseteq \Big(\bigotimes_{\nu' \in \operatorname{sons}(\nu)} \mathscr{U}_{\alpha_{\nu'}} \Big) \otimes \ell^2(\mathscr{I}_{\beta_\nu}), \tag{3.16}$$

and

$$\mathbf{u} \in \Big(\bigotimes_{\nu \in \operatorname{sons}(\nu^*)} \mathscr{U}_{\alpha_\nu} \Big) \otimes \ell^2(\mathscr{I}_{\beta_{\nu^*}}), \tag{3.17}$$

by [61, Lemma 6.28]. Let $\{\mathbf{U}^\nu(\cdot,\dots,\cdot,k_{\eta(\nu)}) : k_{\eta(\nu)} = 1,\dots,r_{\eta(\nu)}\}$ be a basis of $\mathscr{U}_{\alpha_\nu}$, with $\eta(\nu) = \{\alpha_\nu, \alpha_\nu^c\}$. We also set $\mathbf{U}^{\nu^*} = \mathbf{u}$. Now if a node ν has no sons, we choose $\mathbf{C}^\nu = \mathbf{U}^\nu$. For other $\nu \neq \nu^*$, by (3.16) or (3.17), a tensor $\mathbf{C}^\nu$ is obtained by recursive expansion. By construction, the final representation for $\mathbf{u}$ yields a decomposition according to the tree network. □

3.6 Existence of best approximations

We can state the result of Theorem 3.3 differently. Let $\mathscr{H}_{\le\mathfrak{r}} = \mathscr{H}_{\le\mathfrak{r}}(\mathbb{E})$ denote the set of all tensor representable in a given tensor tree network with edges $\mathbb{E}$. For every $\eta \in \mathbb{E}$ let

$$\mathscr{M}^\eta_{\le r_\eta} = \{\mathbf{u} \in \ell^2(\mathscr{I}) : \operatorname{rank}_\eta(\mathbf{u}) \le r_\eta\}.$$

Then Theorem 3.3 states that

$$\mathscr{H}_{\le\mathfrak{r}} = \{\mathbf{u} : \operatorname{rank}_{\mathbb{E}}(\mathbf{u}) \le \mathfrak{r}\} = \bigcap_{\eta\in\mathbb{E}} \mathscr{M}^\eta_{\le r_\eta}. \tag{3.18}$$

Using the singular value decomposition, it is relatively easy to show that for any finite r_η the set $\mathscr{M}^\eta_{\le r_\eta}$ is weakly sequentially compact [130,48,61,132], and for $r_\nu = \infty$, we have $\mathscr{M}^\eta_{\le r_\eta} = \ell^2(\mathscr{I})$. Hence the set $\mathscr{H}_{\le\mathfrak{r}}$ is weakly sequentially closed. Depending on the chosen norm, this is even true in tensor product of Banach spaces [48]. A standard consequence in reflexive Banach spaces like $\ell^2(\mathscr{I})$ (see, e.g., [143]) is the following.

Theorem 3.4 *Every $\mathbf{u} \in \ell^2(\mathscr{I})$ admits a best approximation in $\mathscr{H}_{\le\mathfrak{r}}$.*

For matrices we know that truncation of the singular value decomposition to rank r yields the best rank-r approximation of that matrix. The analogous problem to find a best approximation of tree rank at most $\mathfrak{r}$ for a tensor $\mathbf{u}$, that is, a best approximation in $\mathscr{H}_{\leq \mathfrak{r}}$, has no such clear solution and can be NP-hard to compute [71]. As we are able project onto every set $\mathscr{M}^{\eta}_{\leq r_\eta}$ via SVD, the characterization (3.18) suggests to apply successive projections on these sets to obtain an approximation in $\mathscr{H}_{\leq \mathfrak{r}}$. This works depending on the order of these projections, and is called hierarchical singular value truncation.

3.7 Hierarchical singular value decomposition and truncation

The bases of subspaces considered in the explicit construction used to prove Theorem 3.3 can be chosen arbitrary. When the left singular vectors of $\mathbf{M}_\eta(\mathbf{u})$ are chosen, the corresponding decomposition $\mathbf{u} = \tau_{\mathfrak{r}}(\mathbf{C}^1, \dots, \mathbf{C}^D)$ is called the *hierarchical singular value decomposition* (HSVD) with respect to the tree network with effective edges $\mathbb{E}$. It was first considered in [36] for the Tucker format, later in [56] for the HT and in [105, 107] for the TT format. It was also introduced before in physics for the matrix product representation [136]. The HSVD can be used to obtain low-rank approximations in the tree network. This procedure is called *HSVD truncation*.

Most technical details will be omitted. In particular, we do not describe how to practically compute an exact HSVD representation. For an arbitrary tensor this is typically prohibitively expensive and is therefore avoided in numerical tensor calculus. However, for $\mathbf{u} = \tau_{\mathfrak{r}}(\tilde{\mathbf{C}}^1, \dots \tilde{\mathbf{C}}^D)$ already given in the tree tensor network format, the procedure is quite efficient. The basic idea is as follows. One changes the components from leaves to root to encode *some* orthonormal bases in every node except ν^*, using e.g., QR decompositions that operate only on (matrix reshapes of) the component tensors. Afterwards, it is possible to install HOSV bases from root to leaves using only SVDs on component tensors. Many details are provided in [61].

In the following we assume that $\mathbf{u}$ has tree rank $\mathfrak{s}$ and $\mathbf{u} = \tau_{\mathfrak{s}}(\mathbf{C}^1, \dots, \mathbf{C}^D) \in \mathscr{H}_{\leq \mathfrak{s}}(\mathbb{E})$ is an HSVD representation. Let $\mathfrak{r} \leq \mathfrak{s}$ be given. We consider here the case that all r_η are finite. An HSVD truncation of $\mathbf{u}$ to $\mathscr{H}_{\leq \mathfrak{r}}$ can be derived as follows. Let

$$\mathbf{M}_\eta(\mathbf{u}) = \mathbf{U}^\eta \Sigma^\eta (\mathbf{V}^\eta)^T$$

be the SVD of $\mathbf{M}_\eta$, with $\Sigma^\eta = \operatorname{diag}(\sigma^\eta_1(\mathbf{u}), \sigma^\eta_2(\mathbf{u}), \dots)$ such that $\sigma^\eta_1(\mathbf{u}) \geq \sigma^\eta_2(\mathbf{u}) \geq \cdots \geq 0$. The truncation of a single $\mathbf{M}_\eta(\mathbf{u})$ to rank r_η can be achieved by applying the orthogonal projection

$$\mathbf{P}_{\eta, r_\eta} = \tilde{\mathbf{P}}_{[\eta], r_\mu} \otimes \mathrm{Id}_{[\eta]^c} : \ell^2(\mathscr{I}) \to \mathscr{M}^{\eta}_{\leq r_\eta}, \tag{3.19}$$

where $\tilde{\mathbf{P}}_{[\eta], r_\mu}$ is the orthogonal projection onto the span of r_η dominant left singular vectors of $\mathbf{M}_\eta(\mathbf{u})$. Then $\mathbf{P}_{\eta, r_\eta}(\mathbf{u})$ is the best approximation of $\mathbf{u}$ in the set $\mathscr{M}^{\eta}_{\leq r_\eta}$. Note that $\mathbf{P}_{\eta, r_\eta} = \mathbf{P}_{\eta, r_\eta, \mathbf{u}}$ itself depends on $\mathbf{u}$.

The projections $(\mathbf{P}_{\eta, r_\eta})_{\eta \in \mathbb{E}}$ are now applied consecutively. However, to obtain a result in $\mathscr{H}_{\leq \mathfrak{r}}$, one has to take care of the ordering. Let $\mathbb{T}$ be a generalised partition tree of the tensor network. Considering a $\alpha \in \mathbb{T}$ with son α' we observe the following:

(i) Applying $\mathbf{P}_{\eta,r_\eta}$ with $\eta = \{\alpha, \alpha^c\}$ does not destroy the nestedness property (3.15) at α, simply because the span of only the dominant r_η left singular vectors is a subset of the full span.

(ii) Applying $\mathbf{P}_{\eta',r_{\eta'}}$ with $\eta' = \{\alpha', \alpha'^c\}$ does not increase the rank of $\mathbf{M}_\eta(\mathbf{u})$. This holds because there exists $\beta \subseteq \{1,\dots,d\}$ such that $\mathrm{Id}_{[\eta']^c} \subseteq \mathrm{Id}_{[\eta]^c} \otimes \mathrm{Id}_\beta$. Thus, since $\mathbf{P}_{\eta',r_{\eta'}}$ is of the form (3.19), it only acts as a left multiplication on $\mathbf{M}_\eta(\mathbf{u})$.

Property (ii) by itself implies that the top-to-bottom application of the projections $\mathbf{P}_{\eta,r_\eta}$ will result in a tensor in $\mathscr{H}_{\le\mathfrak{r}}$. Property (i) implies that the procedure can be performed, starting at the root element, by simply setting to zero all entries in the components that relate to deleted basis elements in the current node or its sons, and resizing the tensors accordingly.

Let level η denote the distance of $[\eta]$ in the tree to α^*, and let L be the maximum such level. The described procedure describes an operator

$$\mathbf{H}_\mathfrak{r} : \ell^2(\mathscr{I}) \to \mathscr{H}_{\le\mathfrak{r}}, \quad \mathbf{u} \mapsto \Big(\prod_{\mathrm{level}\,\eta=L} \mathbf{P}_{\eta,r_\eta,\mathbf{u}} \cdots \prod_{\mathrm{level}\,\eta=1} \mathbf{P}_{\eta,r_\eta,\mathbf{u}} \Big)(\mathbf{u}), \tag{3.20}$$

called the *hard thresholding operator*. Remarkably, as the following result shows, it provides a *quasi-optimal projection*. Recall that a best approximation in $\mathscr{H}_{\le\mathfrak{r}}$ exists by Theorem 3.4.

Theorem 3.5 *For any $\mathbf{u} \in \ell^2(\mathscr{I})$, one has*

$$\min_{\mathbf{v}\in\mathscr{H}_{\le\mathfrak{r}}} \|\mathbf{u}-\mathbf{v}\| \le \|\mathbf{u}-\mathbf{H}_\mathfrak{r}(\mathbf{u})\| \le \sqrt{\sum_{\eta\in\mathbb{E}} \sum_{k>r_\eta} \big(\sigma_k^\eta(\mathbf{u})\big)^2} \le \sqrt{|\mathbb{E}|} \min_{\mathbf{v}\in\mathscr{H}_{\le\mathfrak{r}}} \|\mathbf{u}-\mathbf{v}\|.$$

The proof follows more or less immediately along the lines of [56], using the properties $\|\mathbf{u}-P_1P_2\mathbf{u}\|^2 \le \|\mathbf{u}-P_1\mathbf{u}\|^2 + \|\mathbf{u}-P_2\mathbf{u}\|^2$, which holds for any orthogonal projections P_1, P_2, and $\min_{\mathbf{v}\in\mathscr{M}^\eta_{\le r_\eta}} \|\mathbf{u}-\mathbf{v}\| \le \min_{\mathbf{v}\in\mathscr{H}_{\le\mathfrak{r}}} \|\mathbf{u}-\mathbf{v}\|$, which follows from (3.18).

There are sequential versions of hard thresholding operators which traverse the tree in a different ordering, and compute at edge η the best η-rank-r_η approximation of the current iterate by recomputing an SVD. These techniques can be computationally beneficial, but the error cannot be related to the initial HSVD as easily.

3.8 Hierarchical tensors as differentiable manifolds

We now consider geometric properties of $\mathscr{H}_\mathfrak{r} = \mathscr{H}_\mathfrak{r}(\mathbb{E}) = \{\mathbf{u} : \mathrm{rank}_\mathbb{E}(\mathbf{u}) = \mathfrak{r}\}$, that is, $\mathscr{H}_\mathfrak{r} = \bigcap_{\eta\in\mathbb{E}} \mathscr{M}^\eta_{r_\eta}$, where $\mathscr{M}^\eta_{r_\eta}$ is the set of tensors with η-rank exactly r_η. We assume that $\mathfrak{r}$ is such that $\mathscr{H}_\mathfrak{r}$ is not empty. In contrast to the set $\mathscr{H}_{\le\mathfrak{r}}$, it can be shown that $\mathscr{H}_\mathfrak{r}$ is a smooth embedded submanifold if all ranks r_η are finite [75, 134], which enables Riemannian optimisation methods on it as discussed later. This generalises the fact that matrices of fixed rank form smooth manifolds [70].

The cited references consider finite-dimensional tensor product spaces, but the arguments can be transferred to the present separable Hilbert space setting [132], since the concept of submanifolds itself generalises r_η quite straightforwardly, see, e.g., [93,

144]. The case of infinite ranks r_η, however, is more subtle and needs to be treated with care [48,50].

We will demonstrate some essential features using the example of the TT format. Let $\mathfrak{r} = (r_1, \dots, r_{d-1})$ denote finite TT representation ranks. Repeating (3.11), the set $\mathscr{H}_{\leq \mathfrak{r}}$ is then the image of the multilinear map

$$\tau_{\mathrm{TT}} : \mathscr{W} := \mathscr{W}_1 \times \cdots \times \mathscr{W}_d \to \ell^2(\mathscr{I}),$$

where $\mathscr{W}_\nu = \mathbb{R}^{r_{\nu-1}} \otimes \ell^2(\mathscr{I}_\nu) \otimes \mathbb{R}^{r_\nu}$ (with $r_0 = r_d = 1$), and $\mathbf{u} = \tau_{\mathrm{TT}}(\mathbf{G}^1, \dots, \mathbf{G}^d)$ is defined via

$$\mathbf{u}(i_1, \dots, i_d) = \mathbf{G}^1(i_1)\mathbf{G}^2(i_2)\cdots\mathbf{G}^d(i_d). \tag{3.21}$$

The set $\mathscr{W}$ is called the *parameter space* for the TT format with representation rank $\mathfrak{r}$. It is not difficult to deduce from (3.21) that in this case $\mathscr{H}_{\mathfrak{r}} = \tau_{\mathrm{TT}}(\mathscr{W}^*)$, where $\mathscr{W}^*$ is the open and dense subset of parameters $(\mathbf{G}^1, \dots, \mathbf{G}^d)$ for which the embeddings (reshapes) of every $\mathbf{G}^\nu$ into the matrix spaces $\mathbb{R}^{r_{\nu-1}} \otimes (\ell^2(\mathscr{I}_\mu) \otimes \mathbb{R}^{r_\nu})$, respectively $(\mathbb{R}^{r_{\nu-1}} \otimes \ell^2(\mathscr{I}_\nu)) \otimes \mathbb{R}^{r_\nu}$, have full possible rank $r_{\nu-1}$, respectively r_ν. Since τ_{TT} is continuous, this also shows that $\mathscr{H}_{\leq \mathfrak{r}}$ is the closure of $\mathscr{H}_{\mathfrak{r}}$ in $\ell^2(\mathscr{I})$.

A key point that has not been emphasized so far is that the representation (3.21) is by no means unique. We can replace it with

$$\mathbf{u}(i_1, \dots, i_d) = \left[\mathbf{G}^1(i_1)\mathbf{A}^1\right]\left[(\mathbf{A}^1)^{-1}\mathbf{G}^2(i_2)\mathbf{A}^2\right]\cdots\left[(\mathbf{A}^{d-1})^{-1}\mathbf{G}^d(i_d)\right], \tag{3.22}$$

with invertible matrices $\mathbf{A}^\nu$, which yields new components $\tilde{\mathbf{G}}^\nu$ representing the same tensor. This kind of nonuniqueness occurs in all tree tensor networks and reflects the fact that in all except one node only the subspaces are important, not the concrete choice of basis. A central issue in understanding the geometry of tree representations is to remove these redundancies.

A classical approach, pursued in [134,132], is the introduction of equivalence classes in the parameter space. To this end, we interpret the transformation (3.22) as a left action of the Lie group $\mathscr{G}$ of regular matrix tuples $(\mathbf{A}^1, \dots, \mathbf{A}^{d-1})$ on the regular parameter space $\mathscr{W}^*$. The parameters in an orbit $\mathscr{G} \circ (\mathbf{G}^1, \dots, \mathbf{G}^d)$ lead to the same tensor and are called *equvialent*. Using simple matrix techniques one can show that this is the only kind of nonuniqueness that occurs. Hence we can identify $\mathscr{H}_{\mathfrak{r}}$ with the quotient $\mathscr{W}^*/\mathscr{G}$. Since $\mathscr{G}$ acts freely and properly on $\mathscr{W}^*$, the quotient admits a unique manifold structure such that the canonical mapping $\mathscr{W}^* \to \mathscr{W}^*/\mathscr{G}$ is a submersion. One now has to show that the induced mapping $\hat{\tau}_{\mathrm{TT}}$ from $\mathscr{W}^*/\mathscr{G}$ to $\ell^2(\mathscr{I})$ is an embedding to conclude that its image $\mathscr{H}_{\mathfrak{r}}$ is an embedded submanifold. The construction can be extended to general tree tensor networks.

The *tangent space* $\mathscr{T}_{\mathbf{u}}\mathscr{H}_{\mathfrak{r}}$ at $\mathbf{u}$, abbreviated by $\mathscr{T}_{\mathbf{u}}$, is of particular importance for optimisation on $\mathscr{H}_{\mathfrak{r}}$. The previous considerations imply that the multilinear map τ_{TT} is a submersion from $\mathscr{W}^*$ to $\mathscr{H}_{\mathfrak{r}}$. Hence the tangent space at $\mathbf{u} = \tau_{\mathrm{TT}}(\mathbf{C}^1, \dots, \mathbf{C}^d)$ is the range of $\tau'_{\mathrm{TT}}(\mathbf{C}^1, \dots, \mathbf{C}^d)$, and by multilinearity, tangent vectors at $\mathbf{u}$ are therefore of the generic form

$$\begin{aligned}\delta\mathbf{u}(i_1, \dots, i_d) = \delta\mathbf{G}^1(i_1)\mathbf{G}(i_2)\cdots\mathbf{G}^d(i_d)\\ + \ \dots \ + \mathbf{G}^1(i_1)\cdots\mathbf{G}^{d-1}(i_{d-1})\delta\mathbf{G}^d(i_d).\end{aligned} \tag{3.23}$$

As a consequence, a tangent vector δu has TT-rank $\mathfrak{s}$ with $s_\nu \le 2r_\nu$.

Since $\tau'_{\mathrm{TT}}(\mathbf{G}^1,\dots,\mathbf{G}^d)$ is not injective (tangentially to the orbit $\mathscr{G}\circ(\mathbf{G}^1,\dots,\mathbf{G}^d)$, the derivative vanishes), the representation (3.23) of tangent vectors cannot be unique. One has to impose *gauge conditions* in form of a *horizontal space*. Typical choices for the TT format are the spaces $\hat{\mathscr{W}}_\nu = \hat{\mathscr{W}}_\nu(\mathbf{G}^\nu)$, $\nu = 1,\dots,d-1$, comprised of $\delta\mathbf{G}^\nu$ satisfying

$$\sum_{k_{\nu-1}=1}^{r_{\nu-1}} \sum_{i_\nu=1}^{n_\nu} \mathbf{G}^\nu(k_{\nu-1},i_\nu,k_\nu)\delta\mathbf{G}^\nu(k_{\nu-1},i_\nu,k_\nu) = 0, \quad k_\nu = 1,\dots,r_\nu.$$

These $\hat{\mathscr{W}}_\nu$ are the orthogonal complements in $\mathscr{W}_\nu$ of the space of $\delta\mathbf{G}^\nu$ for which there exists an invertible $\mathbf{A}^\nu$ such that $\delta\mathbf{G}^\nu(i_\nu) = \mathbf{G}^\nu(i_\nu)\mathbf{A}^\nu$ for all i_ν. This can be used to conclude that every tangent vector of the generic form (3.23) can be uniquely represented such that

$$\delta\mathbf{G}^\nu \in \hat{\mathscr{W}}_\nu, \quad \nu = 1,\dots,d-1, \tag{3.24}$$

in the fact the different contributions then belong to linearly independent subspaces, the details are in [75]. It follows that the derivative $\tau'_{\mathrm{TT}}(\mathbf{G}^1,\dots,\mathbf{G}^d)$ maps the subspace $\hat{\mathscr{W}}_\nu(\mathbf{G}^1)\times\cdots\times\hat{\mathscr{W}}_{d-1}(\mathbf{G}^{d-1})\times\mathscr{W}_d$ of $\mathscr{W}$ bijectively on $\mathscr{T}_{\mathbf{u}}$.[4] In our example, there is no gauge on the component $\mathbf{G}^d$, but with modified gauge spaces, any component could play this role.

The orthogonal projection $\Pi_{T_{\mathbf{u}}}$ onto the tangent space $\mathscr{T}_{\mathbf{u}}$ is computable in a straightforward way if the basis vectors implicitly encoded at nodes $\nu = 1,\dots,d-1$ are orthonormal, which in turn is not difficult to achieve (using QR decomposition from left to right). Then the decomposition of the tangent space induced by (3.23) and the gauge conditions (3.24) is actually orthogonal. Hence the projection on $\mathscr{T}_{\mathbf{u}}$ can be computed by projecting on the different parts. To do that, let $\mathbf{E}_\nu = \mathbf{E}_\nu(\mathbf{G}^1,\dots,\mathbf{G}^d)$ be the linear map $\delta\mathbf{G}^\nu \mapsto \tau_{\mathrm{TT}}(\mathbf{G}^1,\dots,\delta\mathbf{G}^\nu,\dots\mathbf{G}^d)$. Then the components $\delta\mathbf{G}^\nu$ to represent the orthogonal projection of $\mathbf{v}\in\ell^2(\mathscr{I})$ onto $\mathscr{T}_{\mathbf{u}}$ in the form (3.23) are given by

$$\delta\mathbf{G}^\nu = \begin{cases} P_{\hat{\mathscr{W}}_\nu}\mathbf{E}_\nu^+\mathbf{v}, & \nu = 1,\dots,d-1,\\ \mathbf{E}_\nu^T\mathbf{v}, & \nu = d. \end{cases}$$

Here $P_{\hat{\mathscr{W}}_\nu}$ is the orthogonal projection onto the gauge space $\hat{\mathscr{W}}_\nu$, and $\mathbf{E}_\nu^+ = (\mathbf{E}_\nu^T\mathbf{E}_\nu)^{-1}\mathbf{E}_\nu^T$ is the Moore-Penrose inverse of $\mathbf{E}_\nu$. Indeed, the assumption that $\mathbf{u}$ has TT-rank $\mathfrak{r}$ implies that the matrix $\mathbf{E}_\nu^T\mathbf{E}_\nu$ is invertible. At $\nu = d$ it is actually the identity by our assumption that orthonormal bases are encoded in the other nodes. In operator form, the projector $\Pi_{\mathscr{T}_{\mathbf{u}}}$ can then be written as

$$\Pi_{\mathscr{T}_{\mathbf{u}}}\mathbf{v} = \sum_{\nu=1}^{d-1} \mathbf{E}_\nu P_{\hat{\mathscr{W}}_\nu}\mathbf{E}_\nu^+\mathbf{v} + \mathbf{E}_\nu\mathbf{E}_\nu^T\mathbf{v}.$$

The operators $\mathbf{E}_\nu^T$ are simple to implement, since they require only the computation of scalar product of tensors. Furthermore, the inverses $(\mathbf{E}_\nu^T\mathbf{E}_\nu)^{-1}$ are applied only to

[4] Even without assuming our knowledge that $\mathscr{H}_{\mathfrak{r}}$ is an embedded submanifold, these considerations show that τ_{TT} is a smooth map of constant co-rank $r_1^2+\cdots+r_{d-1}^2$ on $\mathscr{W}^*$. This already implies that the image is a locally embedded submanifold [75].

the small component spaces $\mathscr{W}_\nu$. This makes the projection onto the tangent space a flexible and efficient numerical tool for the application of geometric optimisation, see Sec. 4.2. Estimates of the Lipschitz continuity of $\mapsto P_{\mathscr{T}_\mathbf{u}}$ (curvature bounds) are of interest in this context, with upper bounds given in [101,4].

The generalisation of these considerations to arbitrary tree networks is essentially straightforward, but can become notationally quite intricate, see [134] for the HT format.

4 Optimisation with tensor networks and hierarchical tensors and the Dirac-Frenkel variational principle

In this section, our starting point is the abstract optimisation problem of finding

$$\mathbf{u}^* = \operatorname{argmin} J(\mathbf{u}), \quad \mathbf{v} \in \mathscr{A},$$

for a given cost functional $J : \ell^2(\mathscr{I}) \to \mathbb{R}$ and an admissible set $\mathscr{A} \subset \ell^2(\mathscr{I})$.

In general, a minimiser $\mathbf{u}^*$ will not have low hierarchical ranks in any tree tensor network, but we are interested in finding good *low-rank approximations* to $\mathbf{u}^*$. Therefore, let $\mathscr{H}_{\leq \mathfrak{r}}$ denote again a set of tensors representable in a given tree tensor network with corresponding tree ranks at most $\mathfrak{r}$. Then we wish to solve the following tensor product optimisation problem

$$\mathbf{u}_\mathfrak{r} = \operatorname{argmin}\{J(\mathbf{u}) : \mathbf{u} \in \mathscr{C} = \mathscr{A} \cap \mathscr{H}_{\leq \mathfrak{r}}\}. \tag{4.1}$$

By fixing the rank, we have fixed the representation complexity of the approximate solution. In order to achieve a desired accuracy, we have to enrich our model class (systematically). This is often more important, than to find the very result in the fixed model class. The results in [71] show that even the task of finding the best rank approximation is generally NP-hard if $d \geq 3$. We know that in fact there are multiple local minima around the global minimiser. Although the numerical methods in the present chapter represent the fasted and inexpensive approaches to obtain low rank approximation by hierarchical tensors, one should be aware that they are considered as efficient methods to find reasonable approximations.

Typical examples of such optimisation tasks that we have in mind are the following, see also [46,47].

(a) Best rank-$\mathfrak{r}$ approximation in $\ell^2(\mathscr{I})$: for given $\mathbf{v} \in \ell^2(\mathscr{I})$ minimise

$$J(\mathbf{u}) := \|\mathbf{u} - \mathbf{v}\|^2$$

over $\mathscr{A} = \ell^2(\mathscr{I})$. This is the most basic task we encounter in low-rank tensor approximation.

(b) Solving linear operator equations: for elliptic self-adjoint $\mathbf{A} : \ell^2(\mathscr{I}) \to \ell^2(\mathscr{I})$ and $\mathbf{b} \in \ell^2(\mathscr{I})$, we consider $\mathscr{A} := \ell^2(\mathscr{I})$ and

$$J(\mathbf{u}) := \frac{1}{2}\langle \mathbf{A}\mathbf{u}, \mathbf{u}\rangle - \langle \mathbf{b}, \mathbf{u}\rangle \tag{4.2}$$

to solve $\mathbf{Au} = \mathbf{b}$. For nonsymmetric isomorphisms $\mathbf{A}$, one may resort to a least squares formulation

$$J(\mathbf{u}) := \|\mathbf{Au} - \mathbf{b}\|^2, \tag{4.3}$$

The latter approach of minimisation of residual norms also carries over to nonlinear problems.

(c) computing the lowest eigenvalue of symmetric $\mathbf{A}$ by minimisation of the Rayleigh quotient

$$\mathbf{u}^* = \operatorname{argmin}\{J(\mathbf{u}) = \langle \mathbf{Au}, \mathbf{u}\rangle : \|\mathbf{u}\|^2 = 1\}.$$

This approach can be easily extended if one wants to approximate the N lowest eigenvalues and corresponding eigenfunctions simultaneously, see e.g. [84, 40, 104].

For the existence of a minimiser, the weak sequential closedness of the sets $\mathscr{H}_{\leq \mathfrak{r}}$ is crucial. As mentioned before, this property can be violated for tensors described by the canonical format [123, 61], and in general no minimiser exists. However, it does hold for hierarchical tensors $\mathscr{H}_{\leq \mathfrak{r}}$ as was explained in Sec. 3.6. A generalised version of Theorem 3.4 reads as follows.

Theorem 4.1 *Let J be strongly convex over $\ell^2(\mathscr{I})$, and let $\mathscr{A} \subset \ell^2(\mathscr{I})$ be weakly sequentially closed. Then J attains its minimum on $\mathscr{C} = \mathscr{A} \cap \mathscr{H}_{\leq \mathfrak{r}}$.*

As $\mathbf{A}$ is assumed elliptic in example (b) (in the nonsymmetric case this means that the singular values of $\mathbf{A}$ are bounded below), the function J is strongly convex, and one obtains well-posedness of these minimisation problems (this contains (a) as special case).

Since in case (c) the corresponding set $\mathscr{A}$ (the unit sphere) is not weakly closed, such simple arguments do not apply there.

4.1 Alternating linear scheme

We are interested in finding a minimiser, or even less ambitiously, we want to improve the cost functional along our model class when the admissible set is $\mathscr{A} = \ell^2(\mathscr{I})$.

A straightforward approach which suggests itself in view of the multilinearity of $\tau_{\mathrm{TT}}(\mathbf{C}^1, \dots, \mathbf{C}^D)$ is block coordinate descent (BCD). For the task of finding the best rank-$\mathfrak{r}$ approximation this approach is classical and called *alternating least squares* (ALS), because the optimal choice of a single block is obtained from a least squares problem. For more general quadratic optimisation problems we refer to BCD methods as *alternating linear schemes*.

The idea is to iteratively fix all components $\mathbf{C}^\nu$ except one. The restriction $\mathbf{C}^\nu \mapsto \tau_{\mathfrak{r}}(\mathbf{C}^1, \dots, \mathbf{C}^D)$ is linear. Thus for quadratic J we obtain again a quadratic optimisation problem for the unknown component $\mathbf{C}^\nu$, which is of much smaller dimension than the ambient space $\ell^2(\mathscr{I})$. Generically, there exist unique solutions of the restricted problems.

In this way, the nonlinearity imposed by the model class is circumvented at the price of possibly making very little progress in each step or encountering accumulation points which are not critical points of the problem. Also the convergence analysis is

Algorithm 1: Alternating linear scheme

```
while not converged do
    for ν = 1,...,D do
        C^ν ← argmin_{C^ν} J(τ_𝔱(C^1,...,C^ν,...,C^D))
    end
end
```

challenging, as textbook assumptions on BCD methods are typically not met, see [131, 114, 102] for partial results. However, regularisation can cure most convergence issues [142, 133].

In practical computations the abstract description in Algorithm 1 is modified to reorder the tree during the process in such a way that the component to be optimised becomes the root, and all bases encoded in the other nodes are orthonormalized accordingly. This does not affect the generated sequence of tensors [114], but permits much more efficient solution of the local least squares problems. In particular, the condition of the restricted problems is bounded by the condition of the original problem [74]. All contractions required to set up the local linear system for a single component scale only polynomial in r, n, and are hence computable at reasonable costs.

This optimisation procedure for tensor networks is known as the single-site *density matrix renormalization group* (DMRG) algorithm in physics [139]. The two-site DMRG algorithm (modified ALS [74]) has been developed by White [138] for spin chain models. It is a substantial modification of the scheme above, casting neighbouring components $\mathbf{C}^\nu$ and $\mathbf{C}^{\nu+1}$ together in one, which then has to be optimised. Afterwards the result is separated again by an appropriately truncated SVD. This allows an adjustment of representation ranks, but comes at a higher numerical cost. In the numerical analysis community such algorithms have been used in [80, 74, 87, 108, 40, 84].

4.2 Riemannian gradient descent

The Riemannian optimisation framework [1] assumes that the minimiser $\mathbf{u}_\mathfrak{t} \in \mathscr{H}_{\leq \mathfrak{t}}$ of the problem constrained to $\mathscr{H}_{\leq \mathfrak{t}}$ actually belongs to the smooth manifold $\mathscr{H}_\mathfrak{t}$ (cf. Sec. 3.8). For matrix manifolds this is the case if the global minimiser $\mathbf{u}^*$ does not belong to the singular points $\mathscr{H}_{\leq \mathfrak{t}} \setminus \mathscr{H}_\mathfrak{t}$, see [119].

Assuming $\mathbf{u}_\mathfrak{t} \in \mathscr{H}_\mathfrak{t}$, the first-order necessary optimality condition is that the gradient of J at $\mathbf{u}_\mathfrak{t}$ is perpendicular to the tangent space $\mathscr{T}_{\mathbf{u}_\mathfrak{t}} = \mathscr{T}_{\mathbf{u}_\mathfrak{t}} \mathscr{H}_\mathfrak{t}$. Hence a relaxed problem compared to (4.1) consists in finding $\mathbf{u} \in \mathscr{H}_\mathfrak{t}$ such that

$$\langle \nabla J(\mathbf{u}), \delta \mathbf{u} \rangle = 0 \quad \text{for all } \delta \mathbf{u} \in \mathscr{T}_\mathbf{u}, \tag{4.4}$$

where ∇J is the gradient of J. Since $\mathscr{H}_\mathfrak{t}$ is an embedded submanifold, a trivial Riemannian metric is inherited from the ambient space $\ell^2(\mathscr{I})$, and for the *Riemannian gradient* one has $\mathrm{Grad} J(\mathbf{u}) = P_{\mathscr{T}_\mathbf{u}} \nabla J(\mathbf{u})$, which by (4.4) should be driven to zero.

As a relatively general way of treating the above problems, we will consider *projected gradient methods*. In these methods, one performs gradient steps $\mathbf{y}^{n+1} := \mathbf{u}^n - \alpha_m \nabla J(\mathbf{u}^n)$ in the ambient space $\ell^2(\mathscr{I})$. (More generally, one may take *preconditioned* gradient steps, which is not considered for brevity.) For the problems considered above, $\mathbf{y}^{n+1}$ is in principle computable whenever the operator and right hand side are themselves of low rank, which is the case in many applications. The gradient step is followed by a mapping $\mathbf{R} : \ell^2(\mathscr{I}) \to \mathscr{H}_{\leq \mathfrak{r}}$ to get back on the admissible set. The iteration is summarised as follows:

$$\begin{aligned} \mathbf{y}^{n+1} &:= \mathbf{u}^n - \alpha_n \nabla J(\mathbf{u}^n) && \text{(gradient step)}, \\ \mathbf{u}^{n+1} &:= \mathbf{R}(\mathbf{y}^{n+1}) && \text{(projection step)}. \end{aligned}$$

The specification of the above algorithm depends on the step size selection α_n and on the choice of the projection operator $\mathbf{R} : \ell^2(\mathscr{I}) \to \mathscr{H}_{\leq \mathfrak{r}}$.

Let us remark that taking the best approximation

$$\mathbf{R}(\mathbf{y}^{n+1}) := \operatorname{argmin}\{\|\mathbf{y}^{n+1} - \mathbf{z}\| : \mathbf{z} \in \mathscr{H}_{\leq \mathfrak{r}}\}$$

is generally not numerically realisable [71]. A practically feasible choice for the nonlinear projection $\mathbf{R}$ would be the HSVD truncation $\mathbf{H}_{\mathfrak{r}}$ defined in (3.20), which will be considered in Section 6.1.

Supposing that a *retraction* (defined below) is available on the tangent space, a nonlinear projection $\mathbf{R}$ can also be realised in two steps, by first projecting (linearly) onto the tangent space $\mathscr{T}_{\mathbf{u}^n}$ at $\mathbf{u}^n$, and subsequently applying the retraction R:

$$\begin{aligned} \mathbf{z}^{n+1} &:= P_{\mathscr{T}_{\mathbf{u}^n}}\big(\mathbf{u}^n - \alpha_n \nabla J(\mathbf{u}^n)\big) = \mathbf{u}^n - \alpha_n P_{\mathscr{T}_{\mathbf{u}^n}} \nabla J(\mathbf{u}^n) && \text{(projected gradient step)} \\ &=: \mathbf{u}^n + \xi^n, \quad \xi^n \in \mathscr{T}_{\mathbf{u}^n}, \\ \mathbf{u}^{n+1} &:= R(\mathbf{u}^n, \mathbf{z}^{n+1} - \mathbf{u}^n) = R(\mathbf{u}^n, \xi^n) && \text{(retraction step)}. \end{aligned}$$

In the first line we used that $\mathbf{u}^n \in \mathscr{T}_{\mathbf{u}^n}$ (since $\mathscr{H}_{\mathfrak{r}}$ is a cone). This algorithm is called the *Riemannian gradient iteration*.

Retractions and Riemannian gradient iteration have been introduced in [122]. We follow the treatment in the monograph [1]. A *retraction* maps $\mathbf{u} + \xi$, where $\mathbf{u} \in \mathscr{H}_{\mathfrak{r}}$ and $\xi \in \mathscr{T}_{\mathbf{u}}$, smoothly to a point $R(\mathbf{u}, \xi)$ on the manifold such that

$$\|\mathbf{u} + \xi - R(\mathbf{u}, \xi)\| = \mathscr{O}(\|\xi\|^2).$$

Roughly speaking, a retracting is an approximate exponential map on the manifold, which itself satisfies the definition of a retraction, but is in general too expensive to evaluate. Several examples of retractions for hierarchical tensors are known [101, 85, 100].

Let us note that it can in principle occur that an iterate $\mathbf{u}^n$ is of lower rank, that is, $\mathbf{u}^n \in \mathscr{H}_{\mathfrak{s}}$, where $s_\eta < r_\eta$ at least for one $\eta \in \mathbb{E}$. In this case $\mathbf{u}^n \in \mathscr{H}_{\leq \mathfrak{r}}$ is a singular point, and no longer on the manifold $\mathscr{H}_{\mathfrak{r}}$, so the Riemannian gradient algorithm breaks down. Since $\mathscr{H}_{\mathfrak{r}}$ is dense in $\mathscr{H}_{\leq \mathfrak{r}}$, there exists for arbitrary $\varepsilon > 0$ a tensor $\mathbf{u}^n_\varepsilon \in \mathscr{H}_{\mathfrak{r}}$ with $\|\mathbf{u} - \mathbf{u}^n_\varepsilon\| < \varepsilon$. Practically such a regularised $\mathbf{u}^n_\varepsilon$ is not hard to obtain for a chosen $\varepsilon \sim \|\nabla J(\mathbf{u}^n)\|$. Alternatively, the algorithm described above might be regularised in a sense that it automatically avoids being trapped in a singular point [85].

In [119], the Riemannian gradient iteration was extended to closures of matrix manifolds, and convergence results were deduced from the the Łojasiewicz inequality. We expect that these results can be extended to general tensor manifolds of fixed tree rank.

4.3 Dirac-Frenkel variational principle

The first order optimality condition can be considered as the stationary case of a more general time-dependent formulation in the framework of the *Dirac–Frenkel variational principle* [98]. We consider an initial value problem

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{u} = \mathbf{F}(\mathbf{u}), \quad \mathbf{u}(0) = \mathbf{u}_0 \in \mathscr{H}_{\mathfrak{r}}. \tag{4.5}$$

The goal is to approximate the trajectory $\mathbf{u}(t)$ of (4.5), which might not be exactly of low-rank, by a curve $\mathbf{u}_{\mathfrak{r}}(t)$ in $\mathscr{H}_{\mathfrak{r}}$. The best approximation $\mathbf{u}_{\mathfrak{r}}(t) := \operatorname{argmin}_{\mathbf{v}\in\mathscr{H}_{\mathfrak{r}}} \|\mathbf{u}(t) - \mathbf{v}(t)\|$ provides in general no practical solution to the problem, since i) the computation of the exact trajectory is typically infeasible in high-dimensional problems, and ii) it requires the solution of too many best-approximation problems.

The Dirac-Frenkel variational principle [98] determines an approximate trajectory on a given manifold $\mathbf{u}_{\mathfrak{r}}(t) \in \mathscr{H}_{\mathfrak{r}}$ that minimises

$$\left\| \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{u}(t) - \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{u}_{\mathfrak{r}}(t) \right\| \to \min, \quad \mathbf{u}_{\mathfrak{r}}(0) = \mathbf{u}(0),$$

corresponding to the weak formulation $\langle \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{u}_{\mathfrak{r}} - \mathbf{F}(\mathbf{u}_{\mathfrak{r}}), \delta\mathbf{u}\rangle = 0$ for all $\delta\mathbf{u} \in \mathscr{T}_{\mathbf{u}_{\mathfrak{r}}}$.

If the manifold would be a closed linear space, the equations above are simply the corresponding Galerkin equations. Note also that for the gradient in the limiting case $\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{u} = 0$, one obtains the first order condition (4.4). However, this instationary approach applies also to nonsymmetric operators $\mathbf{A} : \ell^2(\mathscr{I}) \to \ell^2(\mathscr{I})$.

Even for the simple differential equation of the form $\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{u}(t) = \mathbf{F}(t)$, with solution $\mathbf{u}(t) = \mathbf{u}(0) + \int_0^t \mathbf{F}(s)ds$, the Dirac-Frenkel principle leads to a coupled nonlinear system of ODEs, which is not always easy to solve. This motivated the development of splitting schemes that integrate the components successively, similarly to ALS [99, 100]. In particular, the splitting is simple to realise for linear differential equations.

When $\mathbf{F}$ is a partial differential operator, the Dirac-Frenkel principle leads to methods for approximating the solutions of instationary PDEs in high dimension by solving nonlinear systems of low-dimensional differential equations on the tensor manifold $\mathscr{H}_{\mathfrak{r}}$. This shares some similarities with the way how Hartree-Fock and time-dependent Hartree-Fock equations for fermions and the Gross-Pitaevskii equation for bosons are derived. The Dirac-Frenkel principle is well-known in molecular quantum dynamics as the multi-configuration time-dependent Hartree method (MCTDH) [98, 14] for the Tucker format. For hierarchical tensors such a method has been formulated in [137, 98]. First convergence results have been obtained in [101, 4]. The more involved case of reflexive Banach space has been considered in [50]. Time evolution of matrix product states (TT format) for spin systems has been considered in detail in [67].

5 Convergence of low-rank approximations

For a tensor of order d with mode sizes n and all hierarchical ranks bounded by r, the hierarchical format has storage complexity $\mathcal{O}(drn+dr^3)$; in the case of the tensor train format, one obtains $\mathcal{O}(dnr^2)$. Similar results hold for operations on these formats: the HSVD, for instance, requires $\mathcal{O}(dr^2n+dr^4)$ or $\mathcal{O}(dnr^3)$ operations, respectively. For small r, one can thus obtain a very strong improvement over the data complexity n^d of the full tensor. In the numerical treatment of PDEs, however, the underlying function spaces require discretisation. In this context, the above complexity considerations are thus to some degree only formal, since d, n, and r cannot be considered as independent parameters.

5.1 Computational complexity

In the context of PDEs, the appropriate question becomes: what is the total complexity for achieving a prescribed accuracy $\varepsilon > 0$ in the relevant function space norm? In this setting, not only the ranks, but also the dimension n of the univariate trial spaces – and in the example of Section 7.1 even the tensor order d – need to be considered as functions of $\varepsilon > 0$. This leads to the fundamental question of appropriate notions of approximability in terms of which one can quantify the dependencies of $d(\varepsilon)$, $n(\varepsilon)$, $r(\varepsilon)$ on ε.

Here, we need to consider hierarchical tensors in infinite-dimensional spaces. Let $D \subset \mathbb{R}^d$ be a tensor product domain, e.g., $D = I_1 \times \cdots \times I_d$ with $I_1, \dots, I_d \subseteq \mathbb{R}$. As we have noted, Hilbert function spaces such as $L^2(D) = \bigotimes_{\mu=1}^d L^2(I_\mu)$ and $H^1(D)$ are, by an appropriate choice of basis, isomorphic to $\ell^2(\mathbb{N}^d) = \bigotimes_{\mu=1}^d \ell^2(\mathbb{N})$. This isomorphism is of Kronecker rank one in the case of $L^2(D)$, that is, elementary tensors are mapped on elementary tensors, which corresponds to the setting of (2.1). However, this is *not* the case for $H^1(D)$, an important issue discussed further in Section 6.3. For our present purposes, we may thus restrict ourselves to approximation of tensors in the high-dimensional sequence space $\ell^2(\mathscr{I})$.

Besides approximability, a further important question is whether approximate solutions for any given ε can be found at reasonable cost. This is provided, for linear operator equations $\mathbf{A}\mathbf{u} = \mathbf{b}$ on $\ell^2(\mathscr{I})$, by the adaptive low-rank method in [8]. Assume that $\mathbf{u} \in \ell^2(\mathscr{I})$ belongs to a subset for which accuracy ε requires at most the maximum hierarchical rank $r(\varepsilon)$ and the maximum mode size $n(\varepsilon)$. For given ε, the method finds $\mathbf{u}_\varepsilon$ in hierarchical format with $\|\mathbf{u}-\mathbf{u}_\varepsilon\|_{\ell^2(\mathscr{I})} \le \varepsilon$, with ranks and mode sizes bounded up to fixed constants by $r(\varepsilon)$ and $n(\varepsilon)$, respectively. In addition, if for instance $\mathbf{A}$ has finite rank and can be applied efficiently to each tensor mode, then the total number of operations required can be bounded by $C(d)\left(d\,r^4(\varepsilon) + d\,r^2(\varepsilon)\,n(\varepsilon)\right)$, with $C(d)$ polynomial in d – in other words, up to $C(d)$ one has the operation complexity of performing the HSVD on the best approximation of accuracy ε. This is shown in [8] for $n(\varepsilon)$ algebraic and $r(\varepsilon)$ polylogarithmic in ε, but analogous results can be derived for algebraically growing $r(\varepsilon)$ as well. Similar estimates, with additional logarithmic factors, are obtained in [6,7] for problems on Sobolev spaces where $\mathbf{A}$ is not of finite rank. This adaptive scheme is based on iterative thresholding, see also Section 6.1.

5.2 Low-rank approximability

Since $n(\varepsilon)$ is strongly tied to the underlying univariate discretisations, let us now consider in more detail when one can expect to have efficient low-rank approximations of solutions, that is, slow growth of $r(\varepsilon)$ as $\varepsilon \to 0$. The HSVD of tensors yields information on the approximation error in ℓ^2 with respect to the hierarchical ranks: as a consequence of Theorem 3.5, the error of best low-rank approximation of $\mathbf{u}$ is controlled by the decay of its hierarchical singular values.

To quantify the sparsity of sequences, we use weak-ℓ^p-norms. For a given sequence $a = (a_k)_{k\in\mathbb{N}} \in \ell^2(\mathbb{N})$, let a_n^* denote the n-th largest of the values $|a_k|$. Then for $p > 0$, the space $\mathrm{w}\ell^p$ is defined as the collection of sequences for which $|a|_{\mathrm{w}\ell^p} := \sup_{n\in\mathbb{N}} n^{1/p} a_n^*$ is finite, and this quantity defines a quasi-norm on $\mathrm{w}\ell^p$ for $0 < p < 1$, and a norm for $p \geq 1$. It is closely related to the ℓ^p-spaces, since for $p < p'$, one has $\|a\|_{\ell^{p'}} \leq |a|_{\mathrm{w}\ell^p} \leq \|a\|_{\ell^p}$.

Algebraic decay of the hierarchical singular values can be quantified in terms of

$$\|\mathbf{u}\|_{\mathrm{w}\ell^p_*} := \max_{\eta\in\mathbb{E}} |\sigma^\eta(\mathbf{u})|_{\mathrm{w}\ell^p}. \tag{5.1}$$

Note that the *p-th Schatten class*, which one obtains by replacing $\mathrm{w}\ell^p_*$ in (5.1) by ℓ^p, is contained in $\mathrm{w}\ell^p_*$. For these spaces, from Theorem 3.5 we obtain the following low-rank approximation error estimate.

Proposition 5.1 *Let $\mathbf{u} \in \mathrm{w}\ell^p_*$ for $0 < p < 2$. Then there exists a tensor $\hat{\mathbf{u}}$ such that*

$$\|\mathbf{u} - \hat{\mathbf{u}}\| \leq C\sqrt{d}\,\|\mathbf{u}\|_{\mathrm{w}\ell^p_*} \big(\max_{\eta\in\mathbb{E}} \mathrm{rank}_\eta(\hat{\mathbf{u}})\big)^{-s} \quad \text{with } s = \frac{1}{p} - \frac{1}{2}.$$

It has been shown in [118] that, for instance, mixed Sobolev spaces are contained in the Schatten classes; we refer to [118] also for more precise formulation and a discussion of the resulting data complexity. However, classical notions of regularity in Sobolev and Besov spaces provide only a partial answer. For these types of regularity, tensor product bases that achieve the optimal approximation rates are already known, and in this regard there is not much room for improvement by low-rank approximation.

A central question is therefore for which problems one can obtain low-rank approximability beyond that guaranteed by regularity. In particular, *under which conditions do assumptions on the low-rank approximability of input data imply that the solution is again of comparable low-rank approximability?*

Instead of using regularity as in [118], one can show error bounds that decay algebraically with respect to the ranks also under quite general conditions if the considered operator has finite ranks [88]. For many problems, one actually observes numerically a more favourable low-rank approximability with superalgebraic or exponential-type decay of singular values. However, known estimates that show such strong decay are tied to specific situations, such as PDEs with finitely many parameters [81, 86, 5] and Poisson-type problems [55, 33].

There are also relevant counterexamples where the ranks required for a certain accuracy grow strongly with respect to d. A variety of such counterexamples originate from ground state computations of quantum lattice systems, such as one- to three-dimensional spin systems, which in many cases exhibit translation symmetries that

allow a precise analysis. There is a number of works on *area laws* in quantum physics, see e.g. [3] and the references given there. One should note that for problems that are not amenable to low-rank approximation, also most alternative methods break down.

6 Iterative thresholding schemes

Let us consider the variational formulation of the original operator equation $\mathbf{u} = \arg\min_{\mathbf{v}\in\ell^2(\mathscr{I})} J(\mathbf{v})$ with J as in (4.2) or (4.3). In the methods we have considered in Section 4, this problem is approached in a manner analogous to Ritz-Galerkin discretisations: one restricts the minimisation to the manifold $\mathscr{H}_\mathfrak{r}$ of hierarchical tensors with given fixed rank $\mathfrak{r}$, or better to its closure $\mathscr{H}_{\leq\mathfrak{r}}$, and attempts to solve such constrained minimisation problems for J. However, since $\mathscr{H}_\mathfrak{r}$ and $\mathscr{H}_{\leq\mathfrak{r}}$ are not convex, there are generally multiple local minima. Roughly speaking, in this approach one has fixed the model class and aims to achieve a certain accuracy within this class.

Instead, one can also first prescribe an accuracy to obtain a convex admissible set $\mathscr{C}_\varepsilon := \{\mathbf{v}\in\ell^2(\mathscr{I}) : \|\mathbf{A}\mathbf{v}-\mathbf{b}\|\leq\varepsilon\}$. Over this admissible set, one may now try to minimise the computational costs. Roughly speaking, we want to minimise the largest hierarchical rank of $\mathbf{v}$. This can be seen as a motivation for the various methods based on rank truncations that we consider in this section. Note that even in the matrix case $d = 2$, the functional $\mathbf{A}\mapsto\operatorname{rank}(\mathbf{A})$ is not convex. The nuclear norm can be regarded as a convex relaxation of this functional, and its minimisation over $\mathscr{C}_\varepsilon$ by proximal gradient techniques leads to soft thresholding iterations as in Section 6.2 below.

6.1 Iterative hard thresholding schemes

Starting from a (preconditioned) gradient step $\mathbf{u}^{n+1} = \mathbf{u}^n - \mathbf{C}_n^{-1}\nabla J(\mathbf{u}^n)$ in the ambient space $\ell^2(\mathscr{I})$, in order to keep our iterates of low rank, we introduce projection or truncation operators $\mathbf{R}_n, \mathbf{T}_n$, realised by hard thresholding (3.20) of the singular values in the HSVD,

$$\mathbf{u}^{n+1} := \mathbf{R}_n\big(\mathbf{u}^n - \mathbf{T}_n[\mathbf{C}_n^{-1}\nabla J(\mathbf{u}^n)]\big). \tag{6.1}$$

If we take $\mathbf{T}_n := \mathbf{I}$ and $\mathbf{R}_n := \mathbf{H}_\mathfrak{r}$ (the HSVD projection (3.20)), this can be considered as an analogue of iterative hard thresholding in compressive sensing [18] and matrix recovery [127,52]. In the context of low-rank approximation, such truncated iterations based on various representation formats have a rather long history, see e.g. [16,81,64, 86,10,8,17].

We consider the choice $\mathbf{T}_n := \mathbf{I}$, and $\mathbf{R}_n := \mathbf{H}_\mathfrak{r}$ in more detail, using the trivial preconditioner $\mathbf{C}_n := \mathbf{I}$. Defining the mapping $\mathbf{B}$ on $\ell^2(\mathscr{I})$ by $\mathbf{B}(\mathbf{u}) := \mathbf{u} - \nabla J(\mathbf{u})$, we then have the iteration

$$\mathbf{y}^{n+1} := \mathbf{B}(\mathbf{u}^n), \quad \mathbf{u}^{n+1} := \mathbf{H}_\mathfrak{r}(\mathbf{y}^{n+1}), \quad n\in\mathbb{N}. \tag{6.2}$$

Let $\mathbf{u}$ be a fixed point of $\mathbf{B}$, that is, a stationary point of J. As a consequence of Theorem 3.5, denoting by $\mathbf{u}_\mathfrak{r}$ the best approximation of $\mathbf{u}$ of ranks $\mathfrak{r}$, we have the quasi-optimality property

$$\|\mathbf{y}^n - \mathbf{H}_\mathfrak{r}(\mathbf{y}^n)\| \leq c_d\|\mathbf{y}^n - \mathbf{u}_\mathfrak{r}\|, \tag{6.3}$$

where $c_d = \sqrt{d-1}$ in the case of tensor trains, and $c_d = \sqrt{2d-3}$ in the case of the hierarchical format. Making use of this property, one can proceed similarly to [17, §4]: since $\mathbf{u} = \mathbf{B}(\mathbf{u})$ and $\mathbf{y}^{n+1} = \mathbf{B}(\mathbf{u}^n)$, and by (6.3),

$$\begin{aligned}\|\mathbf{u}^{n+1} - \mathbf{u}\| &\le \|\mathbf{H}_{\mathfrak{r}}(\mathbf{y}^{n+1}) - \mathbf{y}^{n+1}\| + \|\mathbf{B}(\mathbf{u}^n) - \mathbf{B}(\mathbf{u})\| \\ &\le c_d\|\mathbf{B}(\mathbf{u}^n) - \mathbf{u}_{\mathfrak{r}}\| + \|\mathbf{B}(\mathbf{u}^n) - \mathbf{B}(\mathbf{u})\| \\ &\le c_d\|\mathbf{u} - \mathbf{u}_{\mathfrak{r}}\| + (1+c_d)\|\mathbf{B}(\mathbf{u}^n) - \mathbf{B}(\mathbf{u})\|.\end{aligned}$$

From this, we immediately obtain the following convergence result.

Proposition 6.1 *Let* $\|\mathbf{B}(\mathbf{v}) - \mathbf{B}(\mathbf{w})\| \le \rho\|\mathbf{v} - \mathbf{w}\|$ *for all* $\mathbf{v}, \mathbf{w} \in \ell^2(\mathscr{I})$*, where* $\beta := (1+c_d)\rho < 1$ *with* c_d *as in* (6.3)*. Then for any* $\mathbf{u}^0 \in \ell^2(\mathscr{I})$*,*

$$\|\mathbf{u}^n - \mathbf{u}\| \le \beta^n\|\mathbf{u}^0 - \mathbf{u}\| + \frac{c_d}{1-\beta}\|\mathbf{u} - \mathbf{u}_{\mathfrak{r}}\|. \tag{6.4}$$

We thus obtain $\limsup_n \|\mathbf{u}^n - \mathbf{u}\| \lesssim \|\mathbf{u} - \mathbf{u}_{\mathfrak{r}}\|$; for this we need, however, an extremely restrictive contractivity property for $\mathbf{B}$. For instance, in the case of the least squares problem (4.3), where one has $\mathbf{B} = \mathbf{I} - \omega\mathbf{A}^*\mathbf{A}$ with suitable $\omega > 0$, this amounts to the requirement

$$(1-\delta^2)\|\mathbf{v}\|^2 \le \|\mathbf{A}\mathbf{v}\|^2 \le (1+\delta^2)\|\mathbf{v}\|^2, \quad \mathbf{v} \in \ell^2(\mathscr{I}), \tag{6.5}$$

with $0 < \delta < 1/\sqrt{1+c_d}$, or in other words, $\mathrm{cond}(\mathbf{A}) < \sqrt{1+2/c_d}$.

Note that the above arguments can be applied also in the case of nontrivial preconditioners $\mathbf{C}_n$ in (6.1). Since obtaining such extremely strong preconditioning is essentially as difficult as solving the original problem, the action of $\mathbf{C}_n^{-1}$ will typically need to be realised by another iterative solver, as considered in [17]. The setting of Proposition 6.1 in itself may thus be of most interest when it suffices to have (6.5) only on a small subset of $\ell^2(\mathscr{I})$, as in compressive sensing-type problems.

A more common approach is to take $\mathbf{T}_\mathbf{n} = \mathbf{H}_{\mathfrak{r}_n}$ with each $\mathfrak{r}_n$ adapted to achieve a certain error bound, for instance such that for an $\varepsilon > 0$ each $\mathbf{u}^{n+1} := \mathbf{H}_{\mathfrak{r}_n}(\mathbf{B}(\mathbf{u}^n))$, now with a general mapping $\mathbf{B}$ in (6.2), satisfies $\|\mathbf{u}^{n+1} - \mathbf{B}(\mathbf{u}^n)\| \le \varepsilon$. In this case, in the setting of Proposition 6.1, but assuming only $\rho < 1$ (i.e., contractivity of $\mathbf{B}$), we obtain

$$\|\mathbf{u}^n - \mathbf{u}\| \le \rho^n\|\mathbf{u}^0 - \mathbf{u}\| + \frac{\varepsilon}{1-\rho}. \tag{6.6}$$

Note that one now has a much weaker assumption on $\mathbf{B}$, but in contrast to (6.2) one generally does not obtain information on the ranks of $\mathbf{u}^n$. To enforce convergence to $\mathbf{u}$, the parameter ε needs to be decreased over the course of the iteration.

When one proceeds in this manner, the appropriate choice of these truncation tolerances is crucial: one does not have direct control over the ranks, and they may become very large when ε is chosen too small. A choice of truncation parameters that ensures that the ranks of $\mathbf{u}^n$ remain comparable to those required for the current error $\|\mathbf{u}^n - \mathbf{u}\|$, while maintaining convergence, is a central part of the adaptive method for linear operator equations in [8,6] that has been mentioned in Section 5.

The choice $\mathbf{R}_n = \mathbf{I}$ and $\mathbf{T}_n := \mathbf{H}_{\mathfrak{r}_n}$ leads to the basic concept of the AMEn algorithm [41], although actually a somewhat different componentwise truncation is used for $\mathbf{T}_n$,

and this is combined with componentwise solves as in the ALS scheme. Note that the basic version of this method with $\mathbf{R}_n = \mathbf{I}$, for which the analysis was carried out in [41], increases the ranks of the iterates in every step; the practically realised version in fact also uses for $\mathbf{R}_n$ a particular type of HSVD truncation. Although the theoretically guaranteed error reduction rates depend quite unfavourably on d, this method shows very good performance in practical tests.

6.2 Iterative soft thresholding schemes

Soft thresholding of sequences by applying $s_\kappa(x) := \operatorname{sgn}(x)\max\{|x|-\kappa,0\}$ for a $\kappa > 0$ to each entry is a non-expansive mapping on ℓ^2, cf. [30,34]. A soft thresholding operation S_κ for matrices (and Hilbert-Schmidt operators) can be defined as application of s_κ to the singular values. Then S_κ is *non-expansive* in the Frobenius (or Hilbert-Schmidt) norm [20,9].

On this basis, a non-expansive soft thresholding operation for the rank reduction of hierarchical tensors is constructed in [9] as follows. By $S_{\kappa,\eta}$ we denote soft thresholding applied to the η-matricisation $\mathbf{M}_\eta(\cdot)$, that is, $S_{\kappa,\eta}(\mathbf{v}) = \mathbf{M}_\eta^{-1} \circ S_\kappa \circ \mathbf{M}_\eta(\mathbf{u})$. The soft shrinkage operator $\mathbf{S}_\kappa : \ell^2(\mathscr{I}) \to \ell^2(\mathscr{I})$ is then given as the successive application of this operation to each matricisation, that is,

$$\mathbf{S}_\kappa(\mathbf{v}) := S_{\kappa,\eta_E} \circ \ldots \circ S_{\kappa,\eta_1}(\mathbf{v}), \tag{6.7}$$

where $\eta_1, \ldots, \eta_E$ is an enumeration of the effective edges $\mathbb{E}$. It is easy to see that the operator $\mathbf{S}_\kappa$ defined in (6.7) is non-expansive on $\ell^2(\mathscr{I})$, that is, for any $\mathbf{v}, \mathbf{w} \in \ell^2(\mathscr{I})$ and $\kappa > 0$, one has $\|\mathbf{S}_\kappa(\mathbf{v}) - \mathbf{S}_\kappa(\mathbf{w})\| \le \|\mathbf{v} - \mathbf{w}\|$.

We now consider the composition of $\mathbf{S}_\kappa$ with an arbitrary convergent fixed point iteration with a contractive mapping $\mathbf{B}\colon \ell^2(\mathscr{I}) \to \ell^2(\mathscr{I})$, where $\rho \in (0,1)$ such that

$$\|\mathbf{B}(\mathbf{v}) - \mathbf{B}(\mathbf{w})\| \le \rho\|\mathbf{v} - \mathbf{w}\|, \quad \mathbf{v}, \mathbf{w} \in \ell^2(\mathscr{I}). \tag{6.8}$$

Lemma 6.2 ([9]) *Assuming* (6.8)*, let* $\mathbf{u}$ *be the unique fixed point of* $\mathbf{B}$*. Then for any* $\kappa > 0$*, there exists a uniquely determined* $\mathbf{u}^\kappa \in \ell^2(\mathscr{I})$ *such that* $\mathbf{u}^\kappa = \mathbf{S}_\kappa\big(\mathbf{B}(\mathbf{u}^\kappa)\big)$*, which satisfies*

$$(1+\rho)^{-1}\|\mathbf{S}_\kappa(\mathbf{u}) - \mathbf{u}\| \le \|\mathbf{u}^\kappa - \mathbf{u}\| \le (1-\rho)^{-1}\|\mathbf{S}_\kappa(\mathbf{u}) - \mathbf{u}\|. \tag{6.9}$$

Let $\mathbf{u}^0 \in \ell^2(\mathscr{I})$*, then* $\|\mathbf{u}^n - \mathbf{u}^\kappa\| \le \rho^n\|\mathbf{u}^0 - \mathbf{u}^\kappa\|$ *for* $\mathbf{u}^{n+1} := \mathbf{S}_\kappa\big(\mathbf{B}(\mathbf{u}^n)\big)$.

For fixed κ, the thresholded gradient iteration thus converges (at the same rate ρ as the unperturbed iteration) to a modified solution $\mathbf{u}^\kappa$, and the distance of $\mathbf{u}^\kappa$ to the exact solution $\mathbf{u}$ is proportional to the error of thresholding $\mathbf{u}$. This needs to be contrasted with (6.4) and (6.6) in the case of hard thresholding, where the thresholded iterations are not ensured to converge, but only to enter a neighbourhood of the solution, and properties like (6.9) that establish a relation to best approximation errors are much harder to obtain (for instance, by strong contractivity of $\mathbf{B}$ as in Proposition 6.1).

Here we now consider the particular case of a quadratic minimisation problem (4.2) with symmetric elliptic $\mathbf{A}$, corresponding to a linear operator equation, where $\mathbf{B} =$

$\mathbf{I}-\omega\mathbf{A}$ with a suitable $\omega>0$. For this problem, based on Lemma 6.2, in [9] a linearly convergent iteration of the form $\mathbf{u}^{n+1}:=\mathbf{S}_{\kappa_n}\big(\mathbf{B}(\mathbf{u}^n)\big)$ with $\kappa_n\to 0$ is constructed, where each iterate $\mathbf{u}^n$ is guaranteed to have *quasi-optimal ranks*. More specifically, for instance if $\mathbf{u}$ belongs to $\mathrm{w}\ell^p_*$ as defined in Section 5, then with a constant $C>0$,

$$\|\mathbf{u}_n-\mathbf{u}\|\le Cd^2\|\mathbf{u}\|_{\mathrm{w}\ell^p_*}\big(\max_{\eta\in\mathbb{E}}\operatorname{rank}_\eta(\mathbf{u}_n)\big)^{-s},\quad s=\frac{1}{p}-\frac{1}{2}.$$

An analogous quasi-optimality statement holds in the case of exponential-type decay $\sigma^\eta_k(\mathbf{u})=\mathscr{O}(e^{-ck^\beta})$ with some $c,\beta>0$.

The central issue in achieving these bounds is how to choose κ_n. Clearly, the κ_n need to decrease sufficiently to provide progress of the iteration toward $\mathbf{u}$, but if they decrease too rapidly this can lead to very large tensor ranks of the iterates. As shown in [9], both linear convergence and the above quasi-optimality property hold if one proceeds as follows: whenever $\|\mathbf{u}_{n+1}-\mathbf{u}_n\|\le\frac{1-\rho}{2\|\mathbf{A}\|\rho}\|\mathbf{A}\mathbf{u}_{n+1}-\mathbf{b}\|$ holds, set $\kappa_{n+1}=\frac{1}{2}\kappa_n$; otherwise set $\kappa_{n+1}=\kappa_n$. The resulting procedure is universal in the sense that in order to achieve the stated rank bounds, nothing needs to be known a priori about the low-rank approximability of $\mathbf{u}$.

This method has not been combined with an adaptive choice of discretisation so far, but the asymptotic bounds on the ranks of each iterate that this method provides are somewhat stronger than those in [8,6], in the sense that they do not depend on the low-rank structure of $\mathbf{A}$.

6.3 Sobolev norms and preconditioning

So far $\mathscr{V}=\bigotimes_{\mu=1}^d\mathscr{V}_\mu$ has been assumed to be a Hilbert space with a *cross norm*, that is, $\|\mathbf{u}\|_\mathscr{V}=\|\mathbf{u}_1\|_{\mathscr{V}_1}\cdots\|\mathbf{u}_d\|_{\mathscr{V}_d}$ for all rank-one tensors $\mathbf{u}=\mathbf{u}_1\otimes\cdots\otimes\mathbf{u}_d\in\mathscr{V}$. Examples of such spaces are L^2-spaces, as well as certain mixed Sobolev spaces, over tensor product domains. Indeed, if $\mathscr{V}$ is endowed with a cross norm, by choice of suitable bases for the $\mathscr{V}_\mu$, one obtains an isomorphism $\ell^2(\mathbb{N}^d)\to\mathscr{V}$ of Kronecker rank one.

Unfortunately, Sobolev norms do not have this property. For instance, in the important case of the standard $H^1_0(D)$-norm $\|v\|^2_{H^1_0(D)}=\sum_{\mu=1}^d\|\partial_{x_\mu}v\|^2$ on a tensor product domain with homogeneous Dirichlet boundary data, for instance $D:=(0,1)^d$, this is related to the fact that the Laplacian is not a rank-one operator. Applying the inverse of the homogeneous Dirichlet Laplacian on D in the corresponding eigenfunction basis representation amounts to multiplication by the diagonal operator with entries

$$\pi^{-2}\big(\nu_1^2+\ldots+\nu_d^2\big)^{-1},\quad \nu\in\mathbb{N}^d. \tag{6.10}$$

Since the eigenfunctions are separable, but the tensor (6.10) does not have a finite-rank representation, the inverse of the Laplacian therefore does not have a representation of finite rank either.

It does, however, have efficient low-rank approximations based on *exponential sums* [19]. For instance, it is shown in [33] that if $f\in H^{-1+\delta}(D)$ for $\delta>0$, then for $A:=-\Delta$,

$$\big\|A^{-1}f-\mathscr{E}_r(A)f\big\|_{H^1}\le C\exp\Big(-\frac{\delta\pi}{2}\sqrt{2}\Big)\|f\|_{H^{-1+\delta}}, \tag{6.11}$$

where $C > 0$ and

$$\mathscr{E}_r(A) := \sum_{k=1}^{r} \omega_{r,k} e^{-\alpha_{r,k} A} \tag{6.12}$$

with certain $\omega_{r,k}, \alpha_{r,k} > 0$. Since the operators $e^{t\Delta}$, $t > 0$, are of rank one, this yields almost exponentially convergent rank-r approximations of the inverse Laplacian.

Approximations of the type (6.12) can also be used for preconditioning. They are particularly useful in the context of *diagonal* preconditioners for wavelet representations of elliptic operators, where the diagonal elements have a form analogous to (6.10). In this case, the operator exponentials in an approximation of the form (6.12) reduce to the exponentials of the diagonal entries corresponding to each tensor mode. In contrast to (6.11), however, in this case sequences of the form (6.10) need to be approximated up to a certain *relative* accuracy. As a consequence, the required rank of the exponential sum then also depends on the considered discretisation subspace. This is analysed in detail in [6,7].

On finite-dimensional subspaces, one can also use multilevel preconditioners such as BPX with tensor structure. This has been considered for space-time formulations of parabolic problems in [2]; in the elliptic case, the analysis of BPX – including the question of d-dependence – is still open in this context.

7 Applications

In principle, high-dimensional partial differential equations on product domains can be discretised directly by tensor product basis functions. This is suitable in our first example of uncertainty quantification problems. We also discuss two further examples, one from quantum chemistry and another one from molecular dynamics, where such a direct approach is not adequate. In these applications, certain reformulations that exploit specific features are much better suited, and we describe how our general setting of tensor approximations can be adapted to these cases.

7.1 Uncertainty quantification

We consider linear diffusion problems on a domain $D \subset \mathbb{R}^m$, $m = 1,2,3$, with given parameter-dependent diffusion coefficients $a(x,y)$ for $x \in D$ and $y \in U$, and with the set of parameter values U to be specified. The parameterized problem reads

$$-\nabla_x \cdot \big(a(x,y)\nabla_x u(x,y)\big) = f(x), \quad x \in D\,,\ y \in U,$$

with appropriate boundary conditions, for instance homogeneous Dirichlet conditions $u(x,y) = 0$ for all $y \in U$ and $x \in \partial D$. In our setting, we aim to solve such problems in the Bochner space $\mathscr{H} = L^2(U, H_0^1(D), \mu)$, where μ is an appropriate measure.

Examples of particular parameterizations that arise in deterministic formulations of stochastic problems are the *affine case*

$$a(x,y) = a_0(x) + \sum_{k=1}^{\infty} y_k a_k(x), \tag{7.1}$$

where $U = [-1,1]^{\mathbb{N}}$ and μ the uniform measure on U, and the *lognormal case*

$$a(x,y) = \exp\Big(a_0(x) + \sum_{k=1}^{\infty} y_k a_k(x)\Big), \tag{7.2}$$

where $U = \mathbb{R}^{\mathbb{N}}$ and μ is the tensor product of standard Gaussian measures (so that the y_k correspond to independent identically distributed normal random variables).

In each case, the solution u can be expressed as a tensor product polynomial expansion (also referred to as *polynomial chaos*) of the form $u(x,y) = \sum_k \mathbf{u}(x,k) \prod_{i=1}^{\infty} p_{k_i}(y_i)$, where p_ℓ, $\ell \in \mathbb{N}$, are the univariate polynomials orthonormal with respect to the underlying univariate measure, and the summation over k runs over the finitely supported multi-indices in $\mathbb{N}_0^{\mathbb{N}}$. We refer to [53,54,94,141,13,27,28,121] and the references therein.

In both cases (7.1) and (7.2), due to the Cartesian product structure of U, the underlying energy space $\mathscr{V} \simeq H_0^1(D) \otimes L_2(U,\mu)$ is a (countable) tensor product of Hilbert spaces, endowed with a cross norm. By truncation of the expansions in (7.1), (7.2), one obtains a finite tensor product.

In this form, tensor decompositions can be used for solving these problems, for instance combined with a finite element discretisation of $H_0^1(D)$, cf. [44]. The total solution error is then influenced by the finite element discretisation, the truncation of coefficient expansions and polynomial degrees, and by the tensor approximation ranks. An adaptive scheme that balances these error contributions by a posteriori error estimators, using tensor train representations and ALS for tensor optimisation with increasing ranks after a few iterations, can be found with numerical tests in [45].

7.2 Quantum physics – fermionic systems

The *electronic Schrödinger equation* describes the stationary motion of a non-relativistic quantum mechanical system of N electrons in a field of K classical nuclei of charge $Z_\eta \in \mathbb{N}$ and fixed positions $R_\eta \in \mathbb{R}^3$, $\eta = 1,\dots,K$. It is an operator eigenvalue equation for the Hamilton operator H, given by

$$H := -\frac{1}{2}\sum_{\xi=1}^{N} \Delta_\xi + V_{\text{ext}} + \frac{1}{2}\sum_{\xi=1}^{N}\sum_{\substack{\zeta=1\\ \zeta\neq\xi}}^{N} \frac{1}{|x_\xi - x_\zeta|}, \quad V_{\text{ext}} := -\sum_{\xi=1}^{N}\sum_{\eta=1}^{K} \frac{Z_\eta}{|x_\xi - R_\eta|},$$

which acts on *wave functions* Ψ that depend on the N spatial coordinates $x_\xi \in \mathbb{R}^3$ and on the N spin coordinates $s_\xi \in \mathbb{Z}_2$ of the electrons. By the *Pauli principle*, the wave functions Ψ needs to be antisymmetric with respect to the particle variables, that is, it needs to change sign under exchange of two distinct variable pairs (x_ξ, s_ξ) and (x_ζ, s_ζ), see e.g. [125]. The corresponding space of wave functions is (see e.g. [21])

$$\mathbb{H}_N^1 = \big[H^1(\mathbb{R}^3 \times \mathbb{Z}_2, \mathbb{C})\big]^N \cap \bigwedge_{\xi=1}^{N} L_2(\mathbb{R}^3 \times \mathbb{Z}_2, \mathbb{C}),$$

where the symbol $\wedge$ denotes the antisymmetric tensor product (*exterior product*). For the sake of simplicity, we focus on the approximation of ground states, where one aims to find

$$\Psi_0 = \operatorname{argmin}\big\{\langle \Phi, H\Phi\rangle : \ \langle \Phi, \Phi\rangle = 1, \ \Phi \in \mathbb{H}^1_N\big\}$$

and the corresponding energy $E_0 = \langle \Psi_0, H\Psi_0\rangle$. It is sufficient in the present setting to consider only real-valued functions, that is, $\mathbb{C}$ can be replaced by $\mathbb{R}$.

Discretisations can be constructed based on antisymmetric tensor products of single-particle basis functions, so-called *Slater determinants*. For a given orthonormal one-particle set

$$\{\varphi_\mu : \mu = 1, \dots, d\} \subset H^1(\mathbb{R}^3 \times \mathbb{Z}_2), \tag{7.3}$$

the corresponding Slater determinants $\varphi_{\mu_1} \wedge \cdots \wedge \varphi_{\mu_d}$, $\mu_1 < \cdots < \mu_d$, form an orthonormal basis of a space $\mathscr{V}_N^d$, called the *Full-CI space*. A Ritz-Galerkin approximation to Ψ_0 can then be obtained by minimising over the finite-dimensional subspace $\mathscr{V}_N^d \subset \mathbb{H}^1_N$, which leads to the discretised eigenvalue problem of finding the lowest $E \in \mathbb{R}$ and corresponding $\Psi \in \mathscr{V}_N^d$ such that

$$\langle \Phi, H\Psi\rangle = E\langle \Phi, \Psi\rangle \quad \text{for all } \Phi \in \mathscr{V}_N^d. \tag{7.4}$$

Starting from a single-particle basis (7.3), where d is greater than the number N of electrons, every ordered selection $\nu_1, \dots, \nu_N$ of $N \le d$ indices corresponds to an N-particle Slater determinant $\Psi_{SL}[\nu_1, \dots, \nu_N]$. The index of each such basis function can be encoded by a binary string $\beta = (\beta_1, \dots, \beta_d)$ of length d, where $\beta_i = 1$ if $i \in \{\nu_1, \dots, \nu_N\}$, and $\beta_i = 0$ otherwise. Setting $\mathbf{e}^0 := (1, \ 0)^T$, $\mathbf{e}^1 := (0, \ 1)^T \in \mathbb{R}^2$, the linear mapping defined by

$$\iota : \ \Psi_{SL}[\nu_1, \dots, \nu_N] \ \mapsto \ \mathbf{e}^{\beta_1} \otimes \ldots \otimes \mathbf{e}^{\beta_d} \ \in \ \mathscr{B}_d \ := \bigotimes_{\mu=1}^{d} \mathbb{R}^2$$

is a unitary isomorphism between the *Fock space* $\mathscr{F}_d = \oplus_{M=0}^d \mathscr{V}_M^d$ and $\mathscr{B}_d$. The solution of the discretised N-electron Schrödinger equation (7.4) is an element of $\mathscr{F}_d$, subject to the constraint that it contains only N-particle Slater determinants, which are eigenfunctions of the *particle number operator* P with eigenvalue N.

On $\mathscr{B}_d$ one can apply tensor approximation techniques without having to deal explicitly with the antisymmetry requirement. The representation of the discretised Hamiltonian $\mathbf{H} : \mathscr{B}_d \to \mathscr{B}_d$ is given by $\mathbf{H} = \iota \circ H \circ \iota^\dagger$. For a given particle number N, we have to restrict the eigenvalue problem to the subspace $\ker(\mathbf{P} - N\mathbf{I})$, with the discrete particle number operator $\mathbf{P} = \iota \circ P \circ \iota^\dagger$. For electrically neutral systems, the exact ground state is an N-particle function, and this constraint can be dropped.

The discrete Hamilton operator $\mathbf{H}$ has a canonical tensor product representation in terms of the one- and two-electron integrals. By the *Slater-Condon rules* [69, 125], one finds

$$\mathbf{H} = \sum_{p,q=1}^{d} h_p^q \, \mathbf{a}_p^\dagger \mathbf{a}_q + \sum_{p,q,r,s=1}^{d} g_{r,s}^{p,q} \, \mathbf{a}_r^\dagger \mathbf{a}_s^\dagger \mathbf{a}_p \mathbf{a}_q,$$

where the coefficients h_p^q, $g_{p,q}^{r,s}$ are given by

$$h_q^p = \big\langle \varphi_p, \big\{ \tfrac{1}{2}\Delta + V_{\mathrm{ext}} \big\} \varphi_q \big\rangle, \quad g_{p,q}^{r,s} = \big\langle \varphi_p \overline{\varphi_r}, \big(|\cdot|^{-1} * \varphi_q \overline{\varphi_s} \big) \big\rangle.$$

Here the discrete *annihilation operators* $\mathbf{a}_p$ and *creation operators* $\mathbf{a}_q^\dagger$ can be written as Kronecker products of the 2×2-matrices

$$\mathbf{A} := \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, \quad \mathbf{A}^\dagger = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}, \quad \mathbf{S} := \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \quad \mathbf{I} := \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix},$$

where $\mathbf{a}_p := \mathbf{S}\otimes\ldots\otimes\mathbf{S}\otimes\mathbf{A}\otimes\mathbf{I}\otimes\ldots\otimes\mathbf{I}$ with $\mathbf{A}$ appearing in the p-th position. Note that compared to the dimension 2^d of the ambient space $\mathscr{B}_d$, representation ranks of $\mathbf{H}$ thus scale only as $\mathscr{O}(d^4)$. For further details, see also [126,95] and the references given there.

With the representation of the particle number operator $\mathbf{P} = \sum_{p,q=1}^d \mathbf{a}_p^\dagger \mathbf{a}_q$, finding the ground state of the discretised Schrödinger equation in binary variational form amounts to solving

$$\min_{\mathbf{v}\in\mathscr{B}_d} \big\{ \langle \mathbf{H}\mathbf{v},\mathbf{v}\rangle : \langle \mathbf{v},\mathbf{v}\rangle = 1 \ , \ \mathbf{P}\mathbf{v} = N\mathbf{v} \big\}. \tag{7.5}$$

Treating this problem by hierarchical tensor representations (e.g., by tensor trains, in this context usually referred to as *matrix product states*) for the d-fold tensor product space $\mathscr{B}_d$, one can obtain approximations of the wave function Ψ that provide insight into separation of quantum systems into subsystems and their entanglement. The formulation (7.5) is fundamental in the modern formulation of many-particle quantum mechanics in terms of *second quantization*. For a recent survey of related MPS techniques in physics see [120].

The practical application of the concepts described above in quantum chemistry is challenging due to the high accuracy requirements. For numerical examples, we refer to, e.g., [126,24,140]. The approach can be especially advantageous in the case of strongly correlated problems, such as the dissociation of molecules as considered in [103], which cannot be treated by classical methods such as Coupled Cluster. The tensor structure can also be exploited for the efficient computation of several eigenstates [84,40].

Remark 7.1 Variants of the above binary coding can also be used in a much more general context. This leads to *vector-tensorization* [60,63], in the tensor train context also called *quantized TT representation* [111,78], which can be applied to vectors $\mathbf{x}\in\mathbb{K}^N$, $\mathbb{K}\in\{\mathbb{R},\mathbb{C}\}$, with $N=2^d$ that are identified with tensors $\mathbf{u}\in\bigotimes_{i=1}^d \mathbb{K}^2$. This identification can be realised by writing each index $j\in\{0,\ldots,2^d-1\}$ in its binary representation $j=\sum_{i=0}^{d-1} c_i 2^i$, $c_i\in\{0,1\}$. The identification $j\simeq(c_1,\ldots,c_d)$, $c_i\in\{0,1\}$ defines a tensor $\mathbf{u}$ of order d with entries $\mathbf{u}(c_1,\ldots,c_d) := \mathbf{x}(j)$. In many cases of interest, the hierarchical representations or approximations of these tensors have low ranks. In particular, for polynomials, exponentials, and trigonometric functions, the ranks are bounded independently of the grid size, and almost exponentially convergent approximations can be constructed for functions with isolated singularities [57,77,76]. There is also a relation to multiresolution analysis [79].

7.3 Langevin dynamics and Fokker-Planck equations

Let us consider the Langevin equation, which constitutes a stochastic differential equation (SDE) of the form

$$dx(t) = -\nabla V\big(x(t)\big)\,dt + \sqrt{\frac{2}{\gamma}}dW_t, \quad \gamma = \frac{1}{k_b T}, \quad x(t) \in \mathbb{R}^d, \tag{7.6}$$

where W_t is an d-dimensional Brownian motion, see e.g. [112]. The corresponding Fokker-Planck equation describes the *transition probability*, and is given by

$$\partial_t u(x,t) = Lu(x,t) := \nabla \cdot \big(u(x,t)\nabla V(x)\big) + \frac{1}{\gamma}\Delta u(x,t), \quad u(x,0) = u_0(x).$$

Here the transition probability is the conditional probability density $u(x,t) = p(x,t\,|\,x_0,0)$ for a particle starting at x_0 to be found at time t at point x.

For simplicity, let us assume that $x \in D = [-R,R]^d$ with homogeneous Neumann boundary conditions. Under rather general conditions, the operator L has a discrete spectrum $0 = \lambda_0 \geq \lambda_1 \geq \ldots \lambda_j \geq \ldots$, $\lambda_j \to -\infty$ if $j \to \infty$ and smooth eigenfunctions φ_j, $j \in \mathbb{N}_0$. It is easy to check that $\varphi_0(x) = \frac{1}{Z}e^{-\beta V(x)}$ is an eigenfunction φ_0 for the eigenvalue $\lambda_0 = 0$, with some normalisation constant $\frac{1}{Z}$ satisfying $\int_D \varphi_0(x)\,dx = 1$. Under reasonable conditions [112] it can be shown that φ_0 is the stationary or equilibrium distribution, $\varphi_0(x) = \lim_{t\to\infty} u(x,t)$.

Instead of L, see e.g. [39], we consider the *transfer operator* defined by mapping a given probability density $u_0(x)$ to a density at some time $\tau > 0$,

$$u_0(x) \mapsto T_\tau u_0(x) := u(x,\tau), \quad x \in D := [-R,R]^d.$$

In general T_τ can be defined by a stochastic transition function $p(x,y;\tau)$, which describes the conditional probability of the system travelling from x to y in a finite time step $\tau > 0$. We do not require explicit knowledge of p, but we make use of the fact that it satisfies the *detailed balance condition* $\pi(x)\,p(x,y,\tau) = \pi(y)\,p(y,x,\tau)$, where $\pi := \frac{1}{\varphi_0}$. Then T_τ is self-adjoint with respect to the inner product with weight π,

$$\langle u, v\rangle_\pi := \int_D u(x)\,v(x)\,\pi(x)\,dx,$$

that is, $\langle T_\tau u, v\rangle_\pi = \langle u, T_\tau v\rangle_\pi$. It has the same eigenfunctions φ_j as the Fokker-Planck operator L and eigenvalues $\sigma_j = e^{\lambda_j}$, with $\sigma_j \in [0,1]$, which accumulate at zero. For the description of meta-stable states, we are interested in the first eigenfunctions φ_j, $j = 0,1,\ldots,m$, where the corresponding eigenvalues σ_j of T_τ are close to one. This provides a good approximation of the dynamics after the fast eigenmodes corresponding to $\sigma_j \approx 0$ are damped out.

In contrast to L, the operator T_τ is bounded in $L_2(D)$, and the eigenvalue problem can be tackled by Galerkin methods using a basis Φ_k, $k \in \mathscr{I}$ and the weighted inner product $\langle\cdot,\cdot\rangle_\pi$: with the ansatz $\varphi_j = \sum_k u_{j,k}\Phi_k$, the unknown coefficients $\mathbf{u}$ and approximate eigenvalues σ are solutions of a generalised discrete eigenvalue problem $\mathbf{M}\mathbf{u} = \sigma\,\mathbf{M}^0\mathbf{u}$, where $\mathbf{M}_{k,\ell} = \langle\Phi_k, T_\tau\Phi_\ell\rangle_\pi$ and $\mathbf{M}^0_{k,\ell} = \langle\Phi_k,\Phi_\ell\rangle_\pi$.

We do not have a low-rank representation of the operator T_τ at our disposal. Nevertheless, if sufficiently long trajectories of the SDE (7.6), that is, $x(t_k)$ for $t_k = kh$, $k = 1,\dots,K$, are available, then the matrix entries $\mathbf{M}_{k,\ell}$ and $\mathbf{M}^0_{k,\ell}$ can be computed by Monte Carlo integration.

Typical choices of basis functions Φ_k are, for instance, piecewise constant functions (Markov state models [116]) or Gaussians combined with collocation (diffusion maps [29]). Here we propose a tensor product basis obtained from univariate basis functions $x_i \mapsto \chi_{\mu_i}(x_i)$, $x_i \in \mathbb{R}$, combined with a low-rank tensor representation of basis coefficients.

For instance, combining tensor train (TT) representations with DMRG iteration for finding good low-rank approximations of the eigenfunctions, in preliminary numerical tests we observe that surprisingly small ranks are sufficient to obtain comparable accuracy as with state-of-the-art alternative methods. This will be reported on in more detail in a forthcoming work.

8 Conclusion

In view of the rapidly growing literature on the subject of this article, the overview that we have given here is necessarily incomplete. Still, we would like to mention some further topics of interest:

- *Adaptive sampling techniques* analogous to *adaptive cross approximation* (ACA) [105, 12], which provide powerful tools to recover not only matrices, but also low-rank hierarchical tensors,
- *Tensor completion* or *tensor recovery* [85, 32, 113], the counterpart to matrix recovery in compressive sensing,
- Applications of hierarchical tensors in *machine learning* [25, 26],
- *Greedy methods*, based on successive best rank-one approximations [49, 22],
- *Rank-adaptive* alternating optimisation methods based on local residuals or low-rank approximations of global residuals [41, 40, 84],
- HSVD truncation estimates in L^∞-norm, see [62],
- Optimisation of the *dimension tree* for the hierarchical format [11], which can give substantially more favorable ranks [58].

Some aspects of low-rank approximations can be considered as topics of future research. For instance, so far the exploitation of sparsity of the component tensors has not been addressed. Combination of hierarchical tensor representations with linear transformations of variables (as in *ridge function* approximations) has not been explored so far either.

References

1. Absil, P.-A., Mahony, R., Sepulchre, R.: Optimization Algorithms on Matrix Manifolds. Princeton University Press, Princeton, NJ (2008)
2. Andreev, R., Tobler, C.: Multilevel preconditioning and low-rank tensor iteration for space-time simultaneous discretizations of parabolic PDEs. Numer. Linear Algebra Appl. **22**(2), 317–337 (2015)

3. Arad, I., Kitaev, A., Landau, Z., Vazirani, U.: An area law and sub-exponential algorithm for 1D systems. arXiv:1301.1162 (2013)
4. Arnold, A., Jahnke, T.: On the approximation of high-dimensional differential equations in the hierarchical Tucker format. BIT **54**(2), 305–341 (2014)
5. Bachmayr, M., Cohen, A.: Kolmogorov widths and low-rank approximations of parametric elliptic PDEs. arXiv:1502.03117 (2015)
6. Bachmayr, M., Dahmen, W.: Adaptive low-rank methods: Problems on Sobolev spaces. arXiv:1407.4919 (2014)
7. Bachmayr, M., Dahmen, W.: Adaptive low-rank methods for problems on Sobolev spaces with error control in L_2. To appear in ESAIM: M2AN. (2015)
8. Bachmayr, M., Dahmen, W.: Adaptive near-optimal rank tensor approximation for high-dimensional operator equations. Found. Comput. Math. **15**(4), 839–898 (2015)
9. Bachmayr, M., Schneider, R.: Iterative methods based on soft thresholding of hierarchical tensors. arXiv:1501.07714 (2015)
10. Ballani, J., Grasedyck, L.: A projection method to solve linear systems in tensor format. Numer. Linear Algebra Appl. **20**(1), 27–43 (2013)
11. Ballani, J., Grasedyck, L.: Tree adaptive approximation in the hierarchical tensor format. SIAM J. Sci. Comput. **36**(4), A1415–A1431 (2014)
12. Ballani, J., Grasedyck, L., Kluge, M.: Black box approximation of tensors in hierarchical Tucker format. Linear Algebra Appl. **438**(2), 639–657 (2013)
13. Beck, J., Tempone, R., Nobile, F., Tamellini, L.: On the optimal polynomial approximation of stochastic PDEs by Galerkin and collocation methods. Math. Models Methods Appl. Sci. **22**(9), 1250023, 33 (2012)
14. Beck, M. H., Jäckle, A., Worth, G. A., Meyer, H.-D.: The multiconfiguration time-dependent Hartree (MCTDH) method: a highly efficient algorithm for propagating wavepackets. Phys. Rep. **324**(1), 1 – 105 (2000)
15. Beylkin, G., Mohlenkamp, M. J.: Numerical operator calculus in higher dimensions. Proc. Natl. Acad. Sci. USA **99**(16), 10246–10251 (electronic) (2002)
16. Beylkin, G., Mohlenkamp, M. J.: Algorithms for numerical analysis in high dimensions. SIAM J. Sci. Comput. **26**(6), 2133–2159 (electronic) (2005)
17. Billaud-Friess, M., Nouy, A., Zahm, O.: A tensor approximation method based on ideal minimal residual formulations for the solution of high-dimensional problems. ESAIM Math. Model. Numer. Anal. **48**(6), 1777–1806 (2014)
18. Blumensath, T., Davies, M. E.: Iterative hard thresholding for compressed sensing. Appl. Comput. Harmon. Anal. **27**(3), 265–274 (2009)
19. Braess, D., Hackbusch, W.: On the efficient computation of high-dimensional integrals and the approximation by exponential sums. In: R. DeVore, A. Kunoth (eds.) Multiscale, Nonlinear and Adaptive Approximation, pp. 39–74. Springer Berlin Heidelberg (2009). DOI 10.1007/978-3-642-03413-8_3
20. Cai, J.-F., Candès, E. J., Shen, Z.: A singular value thresholding algorithm for matrix completion. SIAM J. Optim. **20**(4), 1956–1982 (2010)
21. Cancès, E., Defranceschi, M., Kutzelnigg, W., Le Bris, C., Maday, Y.: Handbook of Numerical Analysis, vol. X, chap. Computational Chemistry: A Primer. North-Holland (2003)
22. Cancès, E., Ehrlacher, V., Lelièvre, T.: Convergence of a greedy algorithm for high-dimensional convex nonlinear problems. Math. Models Methods Appl. Sci. **21**(12), 2433–2467 (2011)
23. Carroll, J. D., Chang, J.-J.: Analysis of individual differences in multidimensional scaling via an n-way generalization of "Eckart-Young" decomposition. Psychometrika **35**(3), 283–319 (1970)
24. Chan, G. K.-L., Sharma, S.: The density matrix renormalization group in quantum chemistry. Annu. Rev. Phys. Chem. **62**, 465–481 (2011)
25. Cichocki, A.: Era of big data processing: a new approach via tensor networks and tensor decompositions. arXiv:1403.2048 (2014)
26. Cichocki, A., Mandic, D., De Lathauwer, L., Zhou, G., Zhao, Q., Caiafa, C., Phan, H. A.: Tensor decompositions for signal processing applications: From two-way to multiway component analysis. IEEE Signal Proc. Mag. **32**(2), 145–163 (2015)
27. Cohen, A., DeVore, R., Schwab, C.: Convergence rates of best *N*-term Galerkin approximations for a class of elliptic sPDEs. Found. Comput. Math. **10**(6), 615–646 (2010)
28. Cohen, A., Devore, R., Schwab, C.: Analytic regularity and polynomial approximation of parametric and stochastic elliptic PDE's. Anal. Appl. (Singap.) **9**(1), 11–47 (2011)

29. Coifman, R. R., Kevrekidis, I. G., Lafon, S., Maggioni, M., Nadler, B.: Diffusion maps, reduction coordinates, and low dimensional representation of stochastic systems. Multiscale Model. Simul. **7**(2), 842–864 (2008)
30. Combettes, P. L., Wajs, V. R.: Signal recovery by proximal forward-backward splitting. Multiscale Model. Simul. **4**(4), 1168–1200 (electronic) (2005)
31. Comon, P., Luciani, X., de Almeida, A. L. F.: Tensor decompositions, alternating least squares and other tales. J. Chemometrics **23**(7-8), 393–405 (2009)
32. Da Silva, C., Herrmann, F. J.: Optimization on the hierarchical Tucker manifold—applications to tensor completion. Linear Algebra Appl. **481**, 131–173 (2015)
33. Dahmen, W., DeVore, R., Grasedyck, L., Süli, E.: Tensor-sparsity of solutions to high-dimensional elliptic partial differential equations. Found. Comput. Math. (2015). In press.
34. Daubechies, I., Defrise, M., De Mol, C.: An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. Comm. Pure Appl. Math. **57**(11), 1413–1457 (2004)
35. De Lathauwer, L., Comon, P., De Moor, B., Vandewalle, J.: High-order power method – Application in Independent Component Analysis. In: Proceedings of the 1995 International Symposium on Nonlinear Theory and its Applications (NOLTA'95), pp. 91–96 (1995)
36. De Lathauwer, L., De Moor, B., Vandewalle, J.: A multilinear singular value decomposition. SIAM J. Matrix Anal. Appl. **21**(4), 1253–1278 (electronic) (2000)
37. DeVore, R. A.: Nonlinear approximation. Acta Numer. **7**, 51–150 (1998)
38. Dolgov, S., Khoromskij, B.: Tensor-product approach to global time-space parametric discretization of chemical master equation. Preprint 68/2012, MPI Leipzig (2012)
39. Dolgov, S. V., Khoromskij, B. N., Oseledets, I. V.: Fast solution of parabolic problems in the tensor train/quantized tensor train format with initial application to the Fokker-Planck equation. SIAM J. Sci. Comput. **34**(6), A3016–A3038 (2012)
40. Dolgov, S. V., Khoromskij, B. N., Oseledets, I. V., Savostyanov, D. V.: Computation of extreme eigenvalues in higher dimensions using block tensor train format. Comput. Phys. Commun. **185**(4), 1207–1216 (2014)
41. Dolgov, S. V., Savostyanov, D. V.: Alternating minimal energy methods for linear systems in higher dimensions. SIAM J. Sci. Comput. **36**(5), A2248–A2271 (2014)
42. Eckart, C., Young, G.: The approximation of one matrix by another of lower rank. Psychometrika **1**(3), 211–218 (1936)
43. Edelman, A., Arias, T. A., Smith, S. T.: The geometry of algorithms with orthogonality constraints. SIAM J. Matrix Anal. Appl. **20**(2), 303–353 (1999)
44. Eigel, M., Gittelson, C. J., Schwab, C., Zander, E.: Adaptive stochastic Galerkin FEM. Comput. Methods Appl. Mech. Engrg. **270**, 247–269 (2014)
45. Eigel, M., Pfeffer, M., Schneider, R.: Adaptive stochastic Galerkin FEM with hierarchical tensor representions. WIAS Berlin preprint 2153
46. Espig, M., Hackbusch, W., Handschuh, S., Schneider, R.: Optimization problems in contracted tensor networks. Comput. Vis. Sci. **14**(6), 271–285 (2011)
47. Espig, M., Hackbusch, W., Rohwedder, T., Schneider, R.: Variational calculus with sums of elementary tensors of fixed rank. Numer. Math. **122**(3), 469–488 (2012)
48. Falcó, A., Hackbusch, W.: On minimal subspaces in tensor representations. Found. Comput. Math. **12**(6), 765–803 (2012)
49. Falcó, A., Nouy, A.: Proper generalized decomposition for nonlinear convex problems in tensor Banach spaces. Numer. Math. **121**(3), 503–530 (2012)
50. Falcó, A., Hackbusch W., Nouy, A.: Geometric structures in tensor representations. Preprint 9/2013, MPI Leipzig (2013)
51. Fannes, M., Nachtergaele, B., Werner, R. F.: Finitely correlated states on quantum spin chains. Comm. Math. Phys. **144**(3), 443–490 (1992)
52. Foucart, S., Rauhut, H.: A Mathematical Introduction to Compressive Sensing. Birkhäuser/Springer, New York (2013)
53. Ghanem, R., Spanos, P. D.: Polynomial chaos in stochastic finite elements. J. Appl. Mech. **57**(1), 197–202 (1990)
54. Ghanem, R. G., Spanos, P. D.: Stochastic Finite Elements: A Spectral Approach, second edn. Dover (2007)
55. Grasedyck, L.: Existence and computation of low Kronecker-rank approximations for large linear systems of tensor product structure. Computing **72**(3-4), 247–265 (2004)
56. Grasedyck, L.: Hierarchical singular value decomposition of tensors. SIAM J. Matrix Anal. Appl. **31**(4), 2029–2054 (2009/10)

57. Grasedyck, L.: Polynomial approximation in hierarchical Tucker format by vector-tensorization. DFG SPP 1324 Preprint 43 (2010)
58. Grasedyck, L., Hackbusch, W.: An introduction to hierarchical (H-)rank and TT-rank of tensors with examples. Computational Methods in Applied Mathematics **11**, 291–304 (2011)
59. Grasedyck, L., Kressner, D., Tobler, C.: A literature survey of low-rank tensor approximation techniques. GAMM-Mitt. **36**(1), 53–78 (2013)
60. Hackbusch, W.: Tensorisation of vectors and their efficient convolution. Numer. Math. **119**(3), 465–488 (2011)
61. Hackbusch, W.: Tensor Spaces and Numerical Tensor Calculus. Springer, Heidelberg (2012)
62. Hackbusch, W.: L^∞ estimation of tensor truncations. Numer. Math. **125**(3), 419–440 (2013)
63. Hackbusch, W.: Numerical tensor calculus. Acta Numer. **23**, 651–742 (2014)
64. Hackbusch, W., Khoromskij, B. N., Tyrtyshnikov, E. E.: Approximate iterations for structured matrices. Numer. Math. **109**(3), 365–383 (2008)
65. Hackbusch, W., Kühn, S.: A new scheme for the tensor representation. J. Fourier Anal. Appl. **15**(5), 706–722 (2009)
66. Hackbusch, W., Schneider, R.: Extraction of Quantifiable Information from Complex Systems, chap. Tensor spaces and hierarchical tensor representations, pp. 237–261. Springer (2014)
67. Haegeman, J., Osborne, T. J., Verstraete, F.: Post-matrix product state methods: To tangent space and beyond. Phys. Rev. B **88**, 075133 (2013)
68. Harshman, R. A.: Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multi-modal factor analysis. UCLA Working Papers in Phonetics **16**, 1–84 (1970)
69. Helgaker, T., Jørgensen, P., Olsen, J.: Molecular Electronic-Structure Theory. John Wiley & Sons, Chichester (2000)
70. Helmke, U., Shayman, M. A.: Critical points of matrix least squares distance functions. Linear Algebra Appl. **215**, 1–19 (1995)
71. Hillar, C. J., Lim, L.-H.: Most tensor problems are NP-hard. J. ACM **60**(6), Art. 45, 39 (2013)
72. Hitchcock, F. L.: The expression of a tensor or a polyadic as a sum of products. Journal of Mathematics and Physics **6**, 164–189 (1927)
73. Hitchcock, F. L.: Multiple invariants and generalized rank of a p-way matrix or tensor. Journal of Mathematics and Physics **7**, 39–79 (1927)
74. Holtz, S., Rohwedder, T., Schneider, R.: The alternating linear scheme for tensor optimization in the tensor train format. SIAM J. Sci. Comput. **34**(2), A683–A713 (2012)
75. Holtz, S., Rohwedder, T., Schneider, R.: On manifolds of tensors of fixed TT-rank. Numer. Math. **120**(4), 701–731 (2012)
76. Kazeev, V., Schwab, C.: Quantized tensor-structured finite elements for second-order elliptic PDEs in two dimensions. SAM research report 2015-24, ETH Zürich (2015)
77. Kazeev, V. A.: Quantized tensor structured finite elements for second-order elliptic pdes in two dimensions. Ph.D. thesis, ETH Zürich (2015)
78. Khoromskij, B. N.: $O(d \log N)$-quantics approximation of N-d tensors in high-dimensional numerical modeling. Constr. Approx. **34**(2), 257–280 (2011)
79. Khoromskij, B. N., Miao, S.: Superfast wavelet transform using quantics-TT approximation. I. Application to Haar wavelets. Comput. Methods Appl. Math. **14**(4), 537–553 (2014)
80. Khoromskij, B. N., Oseledets, I. V.: DMRG+QTT approach to computation of the ground state for the molecular Schrödinger operator. Preprint 69/2010, MPI MIS Leipzig (2010)
81. Khoromskij, B. N., Schwab, C.: Tensor-structured Galerkin approximation of parametric and stochastic elliptic PDEs. SIAM J. Sci. Comput. **33**(1), 364–385 (2011)
82. Koch, O., Lubich, C.: Dynamical tensor approximation. SIAM J. Matrix Anal. Appl. **31**(5), 2360–2375 (2010)
83. Kolda, T. G., Bader, B. W.: Tensor decompositions and applications. SIAM Rev. **51**(3), 455–500 (2009)
84. Kressner, D., Steinlechner, M., Uschmajew, A.: Low-rank tensor methods with subspace correction for symmetric eigenvalue problems. SIAM J. Sci. Comput. **36**(5), A2346–A2368 (2014)
85. Kressner, D., Steinlechner, M., Vandereycken, B.: Low-rank tensor completion by Riemannian optimization. BIT **54**(2), 447–468 (2014)
86. Kressner, D., Tobler, C.: Low-rank tensor Krylov subspace methods for parametrized linear systems. SIAM J. Matrix Anal. Appl. **32**(4), 1288–1316 (2011)
87. Kressner, D., Tobler, C.: Preconditioned low-rank methods for high-dimensional elliptic PDE eigenvalue problems. Comput. Methods Appl. Math. **11**(3), 363–381 (2011)

88. Kressner, D., Uschmajew, A.: On low-rank approximability of solutions to high-dimensional operator equations and eigenvalue problems. arXiv:1406.7026 (2014)
89. Kroonenberg, P. M.: Applied Multiway Data Analysis. Wiley-Interscience [John Wiley & Sons], Hoboken, NJ (2008)
90. Kruskal, J. B.: Rank, decomposition, and uniqueness for 3-way and N-way arrays. In: R. Coppi, S. Bolasco (eds.) Multiway data analysis, pp. 7–18. North-Holland, Amsterdam (1989)
91. Landsberg, J. M.: Tensors: Geometry and Applications. American Mathematical Society, Providence, RI (2012)
92. Landsberg, J. M., Qi, Y., Ye, K.: On the geometry of tensor network states. Quantum Inf. Comput. **12**(3-4), 346–354 (2012)
93. Lang, S.: Fundamentals of Differential Geometry. Springer-Verlag, New York (1999)
94. Le Maître, O. P., Knio, O. M.: Spectral Methods for Uncertainty Quantification. Springer, New York (2010)
95. Legeza, Ö., Rohwedder, T., Schneider, R., Szalay, S.: Many-Electron Approaches in Physics, Chemistry and Mathematics, chap. Tensor product approximation (DMRG) and coupled cluster method in quantum chemistry, pp. 53–76. Springer (2014)
96. Lim, L.-H.: Tensors and hypermatrices. In: L. Hogben (ed.) Handbook of Linear Algebra, second edn. CRC Press, Boca Raton, FL (2014)
97. Lim, L.-H., Comon, P.: Nonnegative approximations of nonnegative tensors. J. Chemometrics **23**(7-8), 432–441 (2009)
98. Lubich, C.: From quantum to classical molecular dynamics: reduced models and numerical analysis. European Mathematical Society (EMS), Zürich (2008)
99. Lubich, C., Oseledets, I. V.: A projector-splitting integrator for dynamical low-rank approximation. BIT **54**(1), 171–188 (2014)
100. Lubich, C., Oseledets, I. V., Vandereycken, B.: Time integration of tensor trains. SIAM J. Numer. Anal. **53**(2), 917–941 (2015)
101. Lubich, C., Rohwedder, T., Schneider, R., Vandereycken, B.: Dynamical approximation by hierarchical Tucker and tensor-train tensors. SIAM J. Matrix Anal. Appl. **34**(2), 470–494 (2013)
102. Mohlenkamp, M. J.: Musings on multilinear fitting. Linear Algebra Appl. **438**(2), 834–852 (2013)
103. Murg, V., Verstraete, F., Schneider, R., Nagy, P. R., Legeza, Ö.: Tree tensor network state study of the ionic-neutral curve crossing of LiF. arXiv:1403.0981 (2014)
104. Nüske, F., Schneider, R., Vitalini, F., Noé, F.: A variational approach for approximating the rare-event kinetics of macromolecular systems. Manuscript
105. Oseledets, I., Tyrtyshnikov, E.: TT-cross approximation for multidimensional arrays. Linear Algebra Appl. **432**(1), 70–88 (2010)
106. Oseledets, I. V.: On a new tensor decomposition. Dokl. Akad. Nauk **427**(2), 168–169 (2009). In Russian; English translation in: Dokl. Math. **80**(1), 495–496 (2009)
107. Oseledets, I. V.: Tensor-train decomposition. SIAM J. Sci. Comput. **33**(5), 2295–2317 (2011)
108. Oseledets, I. V., Dolgov, S. V.: Solution of linear systems and matrix inversion in the TT-format. SIAM J. Sci. Comput. **34**(5), A2718–A2739 (2012)
109. Oseledets, I. V., Tyrtyshnikov, E. E.: Breaking the curse of dimensionality, or how to use SVD in many dimensions. SIAM J. Sci. Comput. **31**(5), 3744–3759 (2009)
110. Oseledets, I. V., Tyrtyshnikov, E. E.: Recursive decomposition of multidimensional tensors. Dokl. Akad. Nauk **427**(1), 14–16 (2009). In Russian; English translation in: Dokl. Math. **80**(1), 460–462 (2009)
111. Oseledets, I. V., Tyrtyshnikov, E. E.: Algebraic wavelet transform via quantics tensor train decomposition. SIAM J. Sci. Comput. **33**(3), 1315–1328 (2011)
112. Pavliotis, G. A.: Stochastic Processes and Applications. Diffusion processes, the Fokker-Planck and Langevin equations. Springer, New York (2014)
113. Rauhut, H., Schneider, R., Stojanac, Z.: Low-rank tensor recovery via iterative hard thresholding. In: 10th international conference on Sampling Theory and Applications (SampTA 2013), pp. 21–24 (2013)
114. Rohwedder, T., Uschmajew, A.: On local convergence of alternating schemes for optimization of convex problems in the tensor train format. SIAM J. Numer. Anal. **51**(2), 1134–1162 (2013)
115. Rozza, G.: Separated Representations and PGD-based Model Reduction, chap. Fundamentals of reduced basis method for problems governed by parametrized PDEs and applications, pp. 153–227. Springer, Vienna (2014)
116. Sarich, M., Noé, F., Schütte, C.: On the approximation quality of Markov state models. Multiscale Model. Simul. **8**(4), 1154–1177 (2010)

117. Schmidt, E.: Zur Theorie der linearen und nichtlinearen Integralgleichungen. Math. Ann. **63**(4), 433–476 (1907)
118. Schneider, R., Uschmajew, A.: Approximation rates for the hierarchical tensor format in periodic Sobolev spaces. J. Complexity **30**(2), 56–71 (2014)
119. Schneider, R., Uschmajew, A.: Convergence results for projected line-search methods on varieties of low-rank matrices via Łojasiewicz inequality. SIAM J. Optim. **25**(1), 622–646 (2015)
120. Schollwöck, U.: The density-matrix renormalization group in the age of matrix product states. Ann. Physics **326**(1), 96–192 (2011)
121. Schwab, C., Gittelson, C. J.: Sparse tensor discretizations of high-dimensional parametric and stochastic PDEs. Acta Numer. **20**, 291–467 (2011)
122. Shub, M.: Some remarks on dynamical systems and numerical analysis. In: Dynamical Systems and Partial Differential Equations (Caracas, 1984), pp. 69–91. Univ. Simon Bolivar, Caracas (1986)
123. de Silva, V., Lim, L.-H.: Tensor rank and the ill-posedness of the best low-rank approximation problem. SIAM J. Matrix Anal. Appl. **30**(3), 1084–1127 (2008)
124. Stewart, G. W.: On the early history of the singular value decomposition. SIAM Rev. **35**(4), 551–566 (1993)
125. Szabo, A., Ostlund, N. S.: Modern Quantum Chemistry. Dover, New York (1996)
126. Szalay, S., Pfeffer, M., Murg, V., Barcza, G., Verstraete, F., Schneider, R., Legeza, Ö.: Tensor product methods and entanglement optimization for ab initio quantum tensor product methods and entanglement optimization for ab initio quantum chemistry. arXiv:1412.5829 (2014)
127. Tanner, J., Wei, K.: Normalized iterative hard thresholding for matrix completion. SIAM J. Sci. Comput. **35**(5), S104–S125 (2013)
128. Tobler, C.: Low-rank tensor methods for linear systems and eigenvalue problems. Ph.D. thesis, ETH Zürich (2012)
129. Tucker, L. R.: Some mathematical notes on three-mode factor analysis. Psychometrika **31**(3), 279–311 (1966)
130. Uschmajew, A.: Well-posedness of convex maximization problems on Stiefel manifolds and orthogonal tensor product approximations. Numer. Math. **115**(2), 309–331 (2010)
131. Uschmajew, A.: Local convergence of the alternating least squares algorithm for canonical tensor approximation. SIAM J. Matrix Anal. Appl. **33**(2), 639–652 (2012)
132. Uschmajew, A.: Zur theorie der niedrigrangapproximation in tensorprodukten von hilberträumen. Ph.D. thesis, Technische Universität Berlin (2013). In German
133. Uschmajew, A.: A new convergence proof for the higher-order power method and generalizations. Pac. J. Optim. **11**(2), 309–321 (2015)
134. Uschmajew, A., Vandereycken, B.: The geometry of algorithms using hierarchical tensors. Linear Algebra Appl. **439**(1), 133–166 (2013)
135. Vandereycken, B.: Low-rank matrix completion by Riemannian optimization. SIAM J. Optim. **23**(2), 1214–1236 (2013)
136. Vidal, G.: Efficient classical simulation of slightly entangled quantum computations. Phys. Rev. Lett. **91**(14), 147902 (2003)
137. Wang, H., Thoss, M.: Multilayer formulation of the multiconfiguration time-dependent Hartree theory. J. Chem Phys. **119**(3), 1289–1299 (2003)
138. White, S. R.: Density matrix formulation for quantum renormalization groups. Phys. Rev. Lett. **69**, 2863–2866 (1992)
139. White, S. R.: Density matrix renormalization group algorithms with a single center site. Phys. Rev. B **72**(18), 180403 (2005)
140. Wouters, S., Poelmans, W., Ayers, P. W., Van Neck, D.: CheMPS2: a free open-source spin-adapted implementation of the density matrix renormalization group for ab initio quantum chemistry. Comput. Phys. Commun. **185**(6), 1501–1514 (2014)
141. Xiu, D.: Numerical Methods for Stochastic Computations. A Spectral Method Approach. Princeton University Press, Princeton, NJ (2010)
142. Xu, Y., Yin, W.: A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion. SIAM J. Imaging Sci. **6**(3), 1758–1789 (2013)
143. Zeidler, E.: Nonlinear Functional Analysis and its Applications. III. Springer-Verlag, New York (1985)
144. Zeidler, E.: Nonlinear Functional Analysis and its Applications. IV. Springer-Verlag, New York (1988)