The Alternating Linear Scheme for Tensor Optimization in the Tensor Train Format

Article *in* SIAM Journal on Scientific Computing · January 2011

DOI: 10.1137/100818893

3 authors, including:

Thorsten Rohwedder
Technische Universität Berlin
17 PUBLICATIONS **1,085** CITATIONS
SEE PROFILE

Reinhold Schneider
Technische Universität Berlin
183 PUBLICATIONS **5,955** CITATIONS
SEE PROFILE

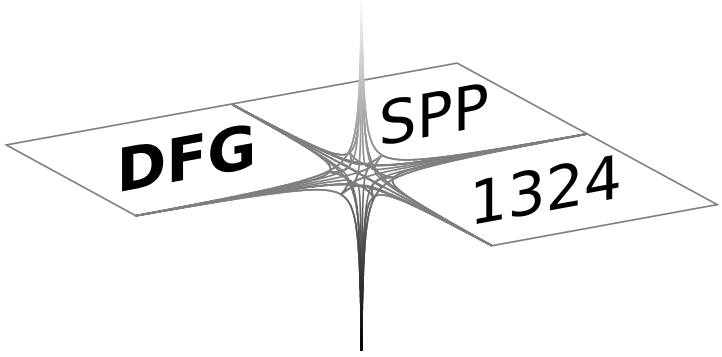
DFG-Schwerpunktprogramm 1324

„Extraktion quantifizierbarer Information aus komplexen Systemen”

The Alternating Linear Scheme for Tensor Optimisation in the TT Format

S. Holtz, T. Rohwedder, R. Schneider

Preprint 71



Edited by

AG Numerik/Optimierung
Fachbereich 12 - Mathematik und Informatik
Philipps-Universität Marburg
Hans-Meerwein-Str.
35032 Marburg

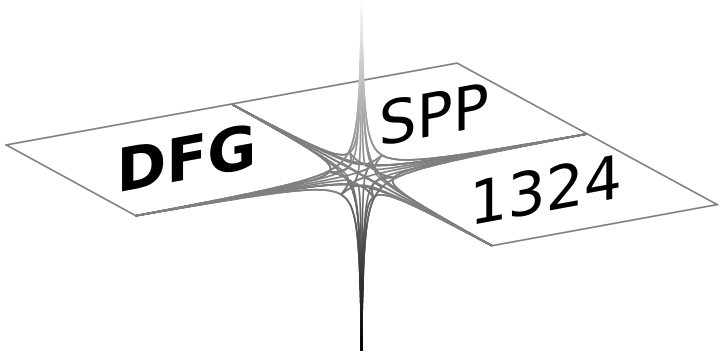
DFG-Schwerpunktprogramm 1324

„Extraktion quantifizierbarer Information aus komplexen Systemen”

The Alternating Linear Scheme for Tensor Optimisation in the TT Format

S. Holtz, T. Rohwedder, R. Schneider

Preprint 71



The consecutive numbering of the publications is determined by their chronological order.

The aim of this preprint series is to make new research rapidly available for scientific discussion. Therefore, the responsibility for the contents is solely due to the authors. The publications will be distributed by the authors.

THE ALTERNATING LINEAR SCHEME FOR TENSOR OPTIMISATION IN THE TT FORMAT

SEBASTIAN HOLTZ, THORSTEN ROHWEDDER AND REINHOLD SCHNEIDER

Abstract. Recent achievements in the field of tensor product approximation [11, 32] provide promising new formats for the representation of tensors in form of tree tensor networks. In contrast to the canonical r-term representation (CANDECOMP, PARAFAC), these new formats provide stable representations [14], while the amount of required data is only slightly larger. The TT format [28, 29, 31], a simple special case of the hierarchical Tucker format [11], is a prototype for such a practical low rank tensor representation, corresponding to the method of matrix product states (MPS) known from quantum information theory.
In this article, we consider two versions of a generalised alternating least square scheme that can be used to treat optimisation tasks in the TT format: A generalisation of the well-known Alternating Least Squares (ALS) algorithm, and a modified approach (MALS) that enables dynamical rank adaptation. For the examples of linear equations and eigenvalue equations, we derive according working equations for the micro-iteration steps and define so-called retraction operators, by the help of which it can easily be shown that many structural properties of the original problems transfer to the micro-iterations. As a practical example of a high-dimensional eigenvalue problem, we take a particular look at the Fock space approach utilised for treatment of the electronic Schrödinger equation and show that the MALS algorithm for this problem corresponds to the Density Matrix Renormalisation Group (DMRG) algorithm, a successful method for the computation of spin chains [40]. We finish with some numerical examples that confirm the positive experiences concerning the stability of the TT decomposition and of ALS and MALS, but also show that in some cases, high TT ranks are required during the iterative approximation of low-rank tensors, showing some potential of improvement.

1. Introduction. The approximation of high dimensional functions, often defined implicitly as the solution of e.g. a partial differential equation, is a major challenge in numerical analysis: Using common discretisation techniques, problems of this kind are usually cast into optimisation problems on corresponding discrete tensor product spaces, say $\bigotimes_{i=1}^{d} \mathbb{R}^n$, where then a linear or nonlinear equation or eigenvalue problem has to be solved. Since the dimension n^d of this space scales exponentially with d, this task is ususally prohibitively expensive for larger d (and/or n) – the well-known *curse of dimensionality*. Instead, approximating the objects under consideration by sums of tensor products of lower rank often provides a powerful tool for tackling such problems.
In the matrix case $d = 2$, the best low rank approximation in the least squares sense is provided by what today is called the singular value decomposition. Although this result is a cornerstone of numerical mathematics (E. Schmidt, 1907, [33]), a direct optimisation of matrices in low rank form [18] has only recently gained interest for model reduction. Unfortunately, Schmidt's results, nowadays usually referred to as Eckhardt-Young theorem [7], does not generalise to dimensions $d > 2$ except in relatively trivial cases. Two distinct ansätze that globalise the SVD are given by

$$U = \sum_{k=1}^{r} \bigotimes_{i=1}^{d} U_{i,k} \quad \text{and} \quad U = \sum_{k_1=1}^{r_1} \dots \sum_{k_d=1}^{r_d} c_{k_1,\dots,k_d} \bigotimes_{i=1}^{d} U_{i,k_i},$$

i.e. the *canonical format* [20], based on a representation by r elementary products of singled valued functions, and the *Tucker format* [37], determining optimal subspaces of dimension r_i in each coordinate direction or mode x_i. While both parametrisations share the property of being highly nonlinear but multi-linear, they are in some sense antipoles exhibiting the advantages and disadvantages one can be faced with

in tensor approximation: In terms of complexity, the canonical format provides an ideal representation, with the number of required parameters depending linearly on the dimension d, the size of the single variable vector spaces n and the number of terms r involved. In contrast to this, representation of a tensor in the Tucker format requires $r^d + nrd$ parameters assuming all $r_i = r$ are equal. Although this usually offers tremendous computational saving of parameters by a factor $\sim (r/n)^d$, it does not prevent from an exponential scaling in degrees of freedom in principle. On the other hand, the treatment of even the simplest problems turns out to be hard to handle practically when approached in the canonical format. Unpredictable redundancies and instabilities lead to typical problems of nonlinear optimisation like non-uniqueness, non-existence etc, so that for the treatment of optimisation tasks in canonical format [6], artificial stabilisation techniques have to be utilised as side conditions [5]. In contrast to this, the Tucker format is mathematically more sound: The Tucker tensors of fixed rank being an embedded manifold [19], it provides a stable parametrisation, reflected in reliable practical computations e.g. in the context of the MCTDHF-method of quantum chemistry [4, 22, 23, 24]. A perfect example for the practical experiences delineated above is the alternating least squares (ALS) method popular for the computation of best approximations in the least squares sense: Fixing all but one component at a time, the parametrisation turns the above multi-linear parametrisations into a series of linear problems for each component. When applied to the canonical format [1], ALS often shows slow or no convergence at all, in particular when higher accuracy is demanded and when the rank r is large; for the Tucker format, its convergence properties are extremely satisfactory [16, 21].

A recent approach proposed by W. Hackbusch et al. [11] combines the advantages of both methods by extending the subspace approximation philosophy to a hierarchical framework, resulting in what is called the Hierarchical Tucker (HT) format. Almost simultaneously, I. Oseledets and E. Tyrtyshnikov [28, 29, 31, 32] presented the *TT format* (abbreviating "tree tensor" or "tensor train"), a powerful special case of HT structure due to its structural simplicity. Although identical approaches have been known for decades in quantum physics, e.g. in quantum dynamics [23, 24], in the context of matrix product states (MPS, [39]) and the DMRG algorithm [34, 40], and have been generalised further to tensor networks [15, 26], these ideas are completely new in the fields of numerics and tensor optimisation. Because the practically motivated perspectives taken in quantum physics are quite different from the systematic approach taken in numerical analysis, a rigorous mathematical treatment of these new tensor representations is still in its early stage. Recently, it has been shown that a stable decomposition providing a quasi-optimal approximation in the ℓ_2-sense can be computed by hierarchical singular value decompositions [9, 29], and that best approximations exist [8]. In a recent publication [14], we proved certain existence and uniqueness statements for TT decompositions, and showed that the manifold of TT tensors of fixed rank shares many of the favourable properties of the corresponding manifold of Tucker tensors proved in [19].
We take these results as a motivation to approach the practical treatment of optimisation problems in the TT format by a suitable generalisation of the ALS algorithm mentioned above, i.e. fixing all indices except one, thus reducing the multi-linear parametrisation in the TT form to a *linear* parametrisation in the free index. Motivated by this fact, we will call the resulting method the *Alternating Linearised Scheme,* again abbreviated by ALS. We also propose a Modified ALS (MALS) algorithm, sim-

ilar to the DMRG algorithm of quantum physics, which facilitates the self-adaption of ranks. A particular emphasis will be laid on the treatment of linear equations and eigenvalue problems. Although we confine our treatment to the case of the TT format for the ease of presentation, many of the methods and results in this paper generalise to the more general HT format without essential difficulties. Note also that in this paper, we restrict our attention to practical aspects of the ALS and MALS algorithms. In a subsequent paper, we will examine the theoretical convergence behaviour of the linearisation approach being the main idea behind these methods, and compare it to an algorithm formulated on the full tangent space as computed in [14].
The paper is organised as follows: We start in Section 2 by introducing our notation for tensors and by explicitly formulating the optimisation problems to be treated. After a short review on the TT format, we discuss the basic ideas of the ALS and MALS methods as relaxation methods, and introduce so-called retraction operators to develop precise algorithmic formulations for ALS and MALS for the TT format. In Section 3, we take a closer look at the important applications of linear equations and the eigenvalue problem. Using the properties of the retraction operators, we find that in this case, ALS and MALS lead to a series of reduced problems of the same kind, inheriting basic features like positive definiteness etc. As an example for a constrained eigenvalue problem, we then apply this approach to formally derive the Fock space ansatz used in quantum chemistry to treat the electronic Schrödinger equation. In this approach, a binary encoding of the full-CI indices leads to a straightforward tensorisation of the full-CI coefficients. Its representation as a TT tensor (matrix product state) and treatment by the modified MALS algorithm then corresponds exactly to the DMRG algorithm for the stationary electronic Schödinger equation as proposed in [3, 41]. By giving this derivation of the DMRG scheme that is to our mind more comprehensible than in the literature, we want to emphasise in particular how the observed connection between tensor product approximation and many particle quantum physics or quantum information theory may provide an ample source for further new developments. In the concluding Section 4, we make some remarks concerning the implementation, and report some of the practical experience we made with ALS and MALS when applied to some sample linear and eigenvalue equations.

2. Tensors, tensor optimisation tasks and the TT format; the ALS and MALS algorithms. To set the stage, this section introduces the objects, problems and algorithms subject to the further treatment in the later sections. We declare some notational conventions used in this work (including the graphical representation already used in [14]) and formulate the optimisation tasks to be treated (Section 2.1). We then review the TT representation for tensors $U \in \mathbb{R}^{n_1 \times \ldots \times n_d}$ (Section 2.2) and finally devise the ALS and MALS algorithms that will be applied to the introduced optimisation tasks in the later sections (Section 2.3).

2.1. Notations and problems to be treated. As in [14], we will treat tensors as multivariate functions, depending on d discrete index sets

$$\mathcal{I}_i = \{1, \ldots, n_i\}, \qquad i \in \{1, \ldots, d\}, \qquad n_i \in \mathbb{N},$$

that is, an order-d tensor is given by

$$U : \mathcal{I}_1 \times \cdots \times \mathcal{I}_d \to \mathbb{R}, \qquad \underline{x} = (x_1, \ldots, x_d) \mapsto U(x_1, \ldots, x_d). \tag{2.1}$$

As in the above, we will in this work only be concerned with real-valued discrete tensors U; the case $\mathbb{K} = \mathbb{C}$ only demands for some obvious modifications. Tensors

from spaces $\mathbb{R}^{n_1\times\ldots\times n_d}$ will be denoted by capital letters $U, B, X, \ldots$. For the special cases $d = 1$ and $d = 2$, corresponding to vectors and matrices, we use bold face fonts and the notation

$$\begin{aligned}\mathbf{v} &:= \big[\mathbf{v}_x\big] := \big[\mathbf{v}_x\big]_{x=1}^{n_1} \in \mathbb{R}^{n_1},\\ \mathbf{A} &:= \big[\mathbf{A}_y^x\big] := \big[\mathbf{A}_y^x\big]_{x=1,y=1}^{n_1,n_2} \in \mathbb{R}^{n_1\times n_2}.\end{aligned}$$

In the context of linear equations, we will sometimes use the (column) vectorisation of a tensor U, denoted in the vein of the above vector notation by the according bold face lower case letter,

$$\mathbf{u} := \big[\mathbf{U}_{x_1,\ldots,x_d}\big] = \big[\mathbf{U}_{x_1,\ldots,x_d}\big]_{\pi(x_1,\ldots,x_d)=1}^{n_1\cdot\ldots\cdot n_d} \in \mathbb{R}^{n_1\cdot\ldots\cdot n_d},$$

where the indices of $\mathbf{u}$ are determined uniquely by a suitable bijection

$$\pi: \ \mathcal{I}_1\times\ldots\times\mathcal{I}_d \ \to \ \{1,\ldots,n_1\cdot\ldots\cdot n_d\}.$$

The transpose of $\mathbf{u}$, a row vector, will sometimes be denoted as $\big[\mathbf{U}^{x_1,\ldots,x_d}\big]$. We will also encounter matrifications of tensors, labelled by subsets of the indices $x_1,\ldots,x_d$ of the tensor (2.1): Again choosing a suitable bijection,

$$\big[\mathbf{U}_{x_1,\ldots,x_i}^{x_{i+1},\ldots,x_d}\big] \ \in \ \mathbb{R}^{(n_1\cdot\ldots\cdot n_i)\times(n_{i+1}\cdot\ldots\cdot n_d)} \tag{2.2}$$

denotes the matrix obtained by taking $x_1,\ldots,x_i$ as row indices and $x_{i+1},\ldots,x_d$ as column indices.

In this publication, we will be concerned with the solution of equations for linear operators A on the tensor space $\mathbb{R}^{n_1\times\ldots\times n_d}$, given by tensors

$$A = A\big(x_1,\ldots,x_n,y_1,\ldots,y_n\big) \in \mathbb{R}^{n_1\times\ldots\times n_d\times n_1\times\ldots\times n_d}, \tag{2.3}$$

and acting on $U\in\mathbb{R}^{n_1\times\ldots\times n_d}$ by the pointwise definition

$$(AU)(y_1,\ldots,y_n) = \sum_{x_1=1}^{n_1}\ldots\sum_{x_d=1}^{n_d} A\big(x_1,\ldots,x_n,y_1,\ldots,y_n\big)U(x_1,\ldots,x_d). \tag{2.4}$$

Using the vectorisation $\mathbf{u} = [\mathbf{U}_{x_1,\ldots,x_n}]$ of U and the matrification $\mathbf{A} = [\mathbf{A}_{y_1,\ldots,y_n}^{x_1,\ldots,x_n}]$ of A, (2.4) can also be computed by re-tensorizing the column vector that results from the matrix-vector multiplication

$$[\mathbf{A}_{y_1,\ldots,y_n}^{x_1,\ldots,x_n}][\mathbf{U}_{x_1,\ldots,x_n}] = \mathbf{A}\mathbf{u},$$

giving a neater, equivalent representation for (2.4). As in this example, we will see that it will often be useful to skip between tensors, denoted in regular letters U, their vectorisation $\mathbf{u} = \big[\mathbf{U}_{x_1,\ldots,x_d}\big]$ and their matrifications $\big[\mathbf{U}_{x_1,\ldots,x_i}^{x_{i+1},\ldots,x_d}\big]$.

Diagrammatic notation. Even with the above notational conventions, a precise description of the tensor objects and operations is often overburden with the indices involved. In our opinion, the description of tensors and their computation is facilitated by the use of diagrammatic representations borrowed from physics and quantum chemistry. This notation is relatively simple, and with some experience one

can decribe most linear tensor operations in a very instructive way since all summations are visualised by edges of a graph. For us, the graphical notation has been of great help in studying and implementing the tensor algorithms, and we strongly recommend this diagrammatic notation for the description of tensor manipulations. In this notation, a tensor U of the above form (2.1) is represented by a dot with d "arms", depicting the d free variables $x_1, \dots, x_d$. The vector and matrix cases $d = 1$ and $d = 2$ as well as the case $d = 5$ are given as example in Figure 2.1. Summation over common indices are symbolised by joining the repective "arms" of the involved tensors: Fig. (d) shows the matrix-vector multiplication

$$\mathbf{A}\mathbf{v} = \big[\mathbf{A}^x_y\big]\big[\mathbf{v}_x\big],$$

while (e) is an example for the more complicated summation

$$W(x_3, x'_3, x_4, x_6) \quad = \quad \sum_{x_1=1}^{n_1} \sum_{x_2=1}^{n_2} \sum_{x_5=1}^{n_5} U(x_1, x_2, x_3, x_4, x_5) V(x_1, x_2, x'_3, x_5, x_6)$$

for U as above (with $d = 5$) and some tensor $V : \mathcal{I}_1 \times \mathcal{I}_2 \times \mathcal{I}_3 \times \mathcal{I}_5 \times \mathcal{I}_6 \to \mathbb{R}$. The diagrams (f), (g) belong to the TT decomposition to be introduced below.

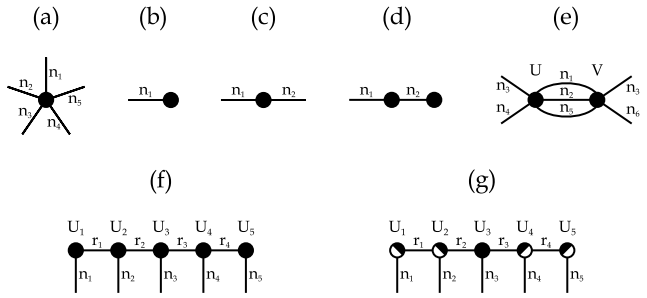


Fig. 2.1: Examples for diagrammatic notation of tensors:
(a) tensor of order 5, (b) tensor of order 1 (vector), (c) tensor of order 2 (matrix),
(d) matrix-vector multiplication, (e) index contraction of tensors U, V (see text),
(f) TT decomposition of (a), (g) orthonormal TT decomposition.

Minimisation tasks on tensor spaces. We will test the TT algorithms TT-GALS and MALS that are main subject of this paper on two very common problems, i.e. the solution of a linear equation

$$AU = B, \tag{2.5}$$

in what $U, B \in \mathbb{R}^{n_1 \times \dots \times n_d}$ are tensors, A is a symmetric positive definite operator mapping $\mathbb{R}^{n_1 \times \dots \times n_d} \to \mathbb{R}^{n_1 \times \dots \times n_d}$, and AU is defined by (2.4), and as a second problem the eigenvalue equation

$$AU = \lambda U, \qquad U \neq 0. \tag{2.6}$$

Both (2.5) and (2.6) are special cases of the minimisation problem for a given functional $\mathcal{J}$, U being the minimiser of

$$\mathcal{J} : \mathbb{R}^{n_1 \times \ldots \times n_d} \ \to \ \mathbb{R}, \quad \mathcal{J}(V) = \frac{1}{2}\langle \mathbf{A}\mathbf{v}, \mathbf{v}\rangle - \langle \mathbf{b}, \mathbf{u}\rangle \tag{2.7}$$

and

$$\mathcal{J} : \mathbb{R}^{n_1 \times \ldots \times n_d} \setminus \{0\} \ \to \ \mathbb{R}, \quad \mathcal{J}(V) = \frac{1}{2}\frac{\langle \mathbf{A}\mathbf{v}, \mathbf{v}\rangle}{\langle \mathbf{v}, \mathbf{v}\rangle}. \tag{2.8}$$

Note that in (2.7) and (2.8), we deliberately define the functional via the matrification $\mathbf{A}$ of the operator A and the vectorisation $\mathbf{v}$ of the tensor V; this viewpoint will be useful later on, when the properties of the resulting vector equations are analysed in Section 3.1.

2.2. The TT decomposition of a tensor. We very shortly review the TT decomposition of tensors, emphasising the points that are of importance for this work. For more detailed information, see [14].
Every tensor $U \in \mathbb{R}^{n_1 \times \ldots \times n_d}$ may be represented exactly in the so-called TT format, i.e. U is decomposed into d component tensors U_i, $i = 1, \ldots, d$ of at most order 3, depending on the variable $x_i \in \mathcal{I}_i$, respectively, and on two auxiliary variables

$$k_{i-1} \in \mathcal{K}_{i-1}, \quad k_i \in \mathcal{K}_i, \qquad \mathcal{K}_i \ := \ \{1, \ldots, r_i\} \text{ for } i = 1, ..., d-1, \quad \mathcal{K}_0 = \mathcal{K}_d = \{1\},$$

connecting the components U_{i-1} and U_i via summation over k_i. Thus,

$$U_i \ \in \ \mathcal{K}_{i-1} \times \mathcal{I}_i \times \mathcal{K}_i \ \to \ \mathbb{R}$$

for all $i \in \{1, \ldots, d\}$. The value of a tensor U at $\underline{x} = (x_1, \ldots, x_d)$ is then given by a successive matrix-vector multiplication using the component tensors,

$$U(\underline{x}) \ = \ \sum_{k_1=1}^{r_1} \ldots \sum_{k_{d-1}=1}^{r_{d-1}} U_1(1, x_1, k_1) \cdot U_2(k_1, x_2, k_2) \cdot \ldots \cdot U_d(k_{d-1}, x_d, 1). \tag{2.9}$$

The numbers r_i are the rank numbers of the TT tensor, giving the TT rank vector $\underline{r} = (r_1, \ldots, r_{d-1})$ that mainly governs the complexity of the TT representation. Note also that for notational convenience, we let $r_0 := r_d := 1$ in the above, so that for all i the values of U_i are given by $U_i(k_{i-1}, x_i, k_i)$ despite the fact that U_1 and U_d depend only on two variables (indices) while all other U_i are order-3-tensors. Cf. also Fig. 2.1 (f) for a depiction of the TT decomposition (2.9).

As in (2.9) above, it is often more instructive to refer to the values $U(\underline{x})$ since the tensor U is therein equipped with the indices or variables $\underline{x} = (x_1, \ldots, x_d)$ it depends on. This is also the case in the probably most comprehensive form for displaying a TT tensor, the *matrix product representation* (called matrix product state (MPS) in quantum physics). This notation makes use of the component function belonging to a component tensor $U_i(k_{i-1}, x_i, k_i)$, defined as

$$\big[(\mathbf{U}_i)_{k_{i-1}}^{k_i}(\cdot)\big] : \quad x_i \ \mapsto \ [U_i(k_{i-1}, x_i, k_i)]_{k_{i-1}=1, k_i=1}^{r_{i-1} \quad r_i} \in \mathbb{R}^{r_{i-1} \times r_i}, \tag{2.10}$$

by which the value of U at $\underline{x} = (x_1, \ldots, x_d)$ can be written as

$$U(\underline{x}) = \big[(\mathbf{U}_1)^{k_1}(x_1)\big] \cdot \ldots \cdot \big[(\mathbf{U}_i)_{k_{i-1}}^{k_i}(x_i)\big] \cdot \ldots \cdot \big[(\mathbf{U}_d)_{k_{d-1}}(x_d)\big]. \tag{2.11}$$

Using the abbreviation

$$\big[\mathbf{U}_{k_{i-1}}^{k_i}\big] = \big[(\mathbf{U}_i)_{k_{i-1}}^{k_i}(\cdot)\big],$$

the tensor U is then represented in the short hand notation by

$$U = \big[\mathbf{U}^{k_1}\big]\big[\mathbf{U}_{k_1}^{k_2}\big] \cdot \ldots \cdot \ldots \cdot \big[\mathbf{U}_{k_{i-1}}^{k_i}\big] \cdot \ldots \cdot \big[\mathbf{U}_{k_{d-2}}^{k_{d-1}}\big]\big[\mathbf{U}_{k_{d-1}}\big]. \tag{2.12}$$

The representation (2.12) is extremely compact and sometimes heavily facilitates the use of TT tensors, but is also rather sloppy and more useful for abbreviation than for pointing to the nature of the tensor decompositions. At several points in this paper we will therefore avoid the notation (2.12) and rather use the more lengthy ones above.

Before we can define the ALS and MALS algorithm, we need some more terminologies and results from [14]: For a component tensor U_i of U, its unfoldings

$$\big[\mathbf{U}_{k_{i-1},x_i}^{k_i}\big] := \big[(\mathbf{U}_i)_{k_{i-1},x_i}^{k_i}\big] \quad \text{and} \quad \big[\mathbf{U}_{k_i,x_i}^{k_{i-1}}\big] := \big[(\mathbf{U}_i)_{k_i,x_i}^{k_{i-1}}\big] \tag{2.13}$$

are called the *left and right unfoldings* of U_i, respectively. U_i is *left resp. right orthogonal* iff the left unfoldings resp. right unfoldings are orthogonal, i.e.

$$\big[\mathbf{U}_{k_i}^{k_{i-1},x_i}\big]\big[\mathbf{U}_{k_{i-1},x_i}^{k_i}\big] = \mathbf{I} \in \mathbb{R}^{r_i \times r_i} \quad \text{resp.} \quad \big[\mathbf{U}_{k_{i-1}}^{k_i,x_i}\big]\big[\mathbf{U}_{k_i,x_i}^{k_{i-1}}\big] = \mathbf{I} \in \mathbb{R}^{r_{i-1} \times r_{i-1}}. \tag{2.14}$$

A TT decomposition is called a *minimal decomposition* if all component functions have full left and right rank,

$$\operatorname{rank}\big[\mathbf{U}_{k_i,x_i}^{k_{i-1}}\big] = r_{i-1}, \quad \operatorname{rank}\big[\mathbf{U}_{k_{i-1},x_i}^{k_i}\big] = r_i \qquad \text{for all} \quad i = 1, \ldots, d,$$

and every such decomposition is optimal in the sense that the ranks r_i equal the respective separation ranks r_i^S of U, while they provide a lower bound for any other TT decomposition. Such a decomposition exists and can be computed by successive singular value decompositions [9, 28], see [14] for the proof.
We finally note for later purposes that, as a straightforward globalisation of Theorem 3.1(b) in [14], for fixed i, the decomposition can be chosen such that the components are left-orthonormal for $j < i$ and right-orthonormal for $j > i$,

$$\big[(\mathbf{U}_j)_{k_j}^{k_{j-1},x_j}\big]\big[(\mathbf{U}_j)_{k_{j-1},x_j}^{k_j}\big] = \mathbf{I} \quad \in \quad \mathbb{R}^{r_j \times r_j} \qquad \text{for} \quad j < i, \tag{2.15}$$

$$\big[(\mathbf{U}_j)_{k_{j-1}}^{k_j,x_j}\big]\big[(\mathbf{U}_j)_{k_j,x_j}^{k_{j-1}}\big] = \mathbf{I} \quad \in \quad \mathbb{R}^{r_{j-1} \times r_{j-1}} \quad \text{for} \quad j > i, \tag{2.16}$$

while the component U_i, sometimes called "core" of the tensor, carries the nonorthogonal share of U. In diagrammatic notation, this may be depicted as in Fig. 2.1 (g), where the black-and-white dots indicate left- and right-orthogonality.

2.3. The ALS and MALS algorithms for the TT format. The TT format represents a tensor U as multi-linear combination of component tensors U_i depending only on few variables (or indices). The alternating linear scheme (ALS) and its modification MALS discussed below are relaxation procedures, where only one resp. two components U_i are optimised at a time while the other components U_j, are fixed. Such relaxation methods are the method of choice in many optimisation and iterative methods, with the Gauß-Seidel iterative method for solving linear equation probably the best known example. In the treatment of multilinear tensor formats as the canonical, the Tucker and also the TT format, relaxation methods have the

attractive property that the parametrisation for the component U_i subject to the momentary micro-iteration enters *linearly* into the tensor product ansatz, so that a global quadratic optimisation problem, for example, then also induces quadratic optimisation problems for the micro-iterations. Convergence to a minimum need not be guaranteed for such relaxation methods, and a rigorous analysis of the convergence properties of the ALS and MALS algorithm will be performed in a forthcoming publication. Here, we will concentrate on the algorithmic aspects of the methods and only note that the iteration is monotone in the sense that the values $\mathcal{J}(U^{(n)})$ decrease during the iteration, see Lemma 2.1.
For each of the two methods, we now give an introduction to the basic ideas of the two algorithms, and then devise a formulation of the algorithms with the help of retraction operators to be defined below.

Alternating Linear Scheme: The first scheme we tested is the Alternating Linear Scheme (ALS), a straightforward generalisation of the well-known Alternating Least Squares algorithm common for approximation of full given tensors. Starting with a TT tensor (2.9) with right-orthonormal component functions $U_2, \dots, U_d$, the first micro-iteration optimises the non-orthonormal component U_1, obtaining a component V_1. Afterwards, the non-orthonormal part (or "core") is moved to the next component function U_2 by performing a QR decomposition

$$[\mathbf{V}^{k_1'}_{k_0,x_1}] = [\tilde{\mathbf{U}}^{k_1}_{k_0,x_1}][\mathbf{R}^{k_1'}_{k_1}] \tag{2.17}$$

of the left unfolding of V_1: A new, left-orthonomal component function $\tilde{U}_1$ is then determined by its left unfolding taken as $\tilde{\mathbf{U}}^{k_1}_{k_0,x_1}$, while the second component U_2 is updated to the tensor $\tilde{U}_2$, determined by its right unfolding chosen as

$$[\tilde{\mathbf{U}}^{x_2,k_2}_{k_1}] := [\mathbf{R}^{k_1'}_{k_1}][\mathbf{U}^{x_2,k_2}_{k_1'}], \tag{2.18}$$

so that $\tilde{U}_2$ now carries the non-orthogonal part of the tensor. This optimisation/orthogonalisation procedure is now performed for $i = 2, \dots, d-1$, moving the non-orthonormal part in each step: A non-orthonormal component U_i is thus optimised at a time, while the remaining components U_j are left-orthonormal for $j < i$ and right-orthonormal for $j > i$ in each step. Such a "half sweep" is depicted in Fig. 2.2 (a). The relaxation procedure is then performeded in the opposite direction, (the "back sweep"), resulting on the whole in an iteration sweep for ALS.

An algorithmic formulation of the ALS scheme can be given in terms of one-component retraction operators $P_{i,1} = P_{i,1,U}$, defined via the current iterate $U = U^{(n)}$ of the ALS scheme. For given U and $i \in \{1, \dots, d\}$,

$$P_{i,1} \;:=\; P_{i,1,U} : \quad \mathbb{R}^{r_{i-1}\times n_i \times r_{i+1}} \;\to\; \bigotimes_{i=1}^{d} \mathbb{R}^{n_i} \tag{2.19}$$

is defined by its action on $V \in \mathbb{R}^{r_{i-1}\times n_i \times r_{i+1}}$, given by

$$P_{i,1}V = [(\mathbf{U}_1)^{k_1}_{k_0}] \cdot .. \cdot [(\mathbf{U}_{i-1})^{k_{i-1}}_{k_{i-2}}][\mathbf{V}^{k_i}_{k_{i-1}}][(\mathbf{U}_{i+1})^{k_{i+1}}_{k_i}] \cdot .. \cdot [(\mathbf{U}_d)^{k_d}_{k_{d-1}}], \tag{2.20}$$

where we used the notation introduced in (2.12). Although $P_{i,1}$ depends on the TT tensor U, this dependency will suppressed as long as there cannot arise any confusion.

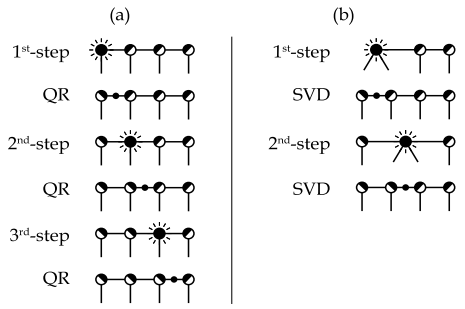


Fig. 2.2: A "half sweep" of the ALS algorithm (a) and of the MALS algorithm (b) for $d = 4$.

Given a functional $\mathcal{J}$ to be optimised, the retraction operators $P_{i,1}$ can now be used to conveniently formulate the ALS relaxation scheme algorithm as given in Fig. 2.3. In the case of linear equations resp. eigenvalue problems, the functional $\mathcal{J}$ is given by (2.7) and (2.8) respectively, and the resulting equations and properties will be analysed in more detail in Section 3 and also serve for numerical examples tested in Section 4. Note also that with some obvious modifications, this algorithm may also be globalised to the hierachical Tucker format.

Require: Functional $\mathcal{J} : \mathbb{R}^{n_1 \times \ldots \times n_d} \to \mathbb{R}$,
Starting vector of component functions $(U_1, \ldots, U_d)$.

Right-orthogonalise the components $U_2, \ldots, U_d$.

while termination criterion is not fulfilled **do**

for $i = 1, \ldots, d-1$ **do**
Compute

$$V_i = \operatorname{argmin}\{\mathcal{J} \circ P_{i,1} \ : \ \mathbb{R}^{r_{i-1} \times n_i \times r_i} \to \mathbb{R}.\} \tag{2.21}$$

Apply the core movement steps (2.17), (2.18) for V_i, giving $\tilde{U}_i, \tilde{U}_{i+1}$.
Set $U_i, U_{i+1} \leftarrow \tilde{U}_i, \tilde{U}_{i+1}$.
end for
Repeat the above loop in reverse order.

end while

Fig. 2.3: The ALS algorithm for TT optimisation problems.

Modified Alternating Linear Scheme: Although our below numerical results suggest that in general, ALS provides a stable, recommendable algorithm, it has two principal disadvantages: Besides the orthogonalisation required in each micro-step to

keep the method stable, ALS carries the main deficit that the ranks cannot easily be adapted to obtain a desired accuracy. We therefore propose a computationally simple Modified Alternating Linear Scheme (MALS) utilizing a greedy strategy which is only slightly more expensive than the ALS for a fixed rank provided the numbers n_i are appropriately small. This scheme is designed exclusively for the TT format, and cannot be easily transferred to the general HT format without deteriorating its efficiency. The idea is to contract two subsequent component functions, say U_i, U_{i+1}, into one compound component function $W \in \mathbb{R}^{r_{i-1}\times n_i \times n_{i-1} \times r_{i+1}}$,

$$W:\ (k_{i-1}, x_i, x_{i+1}, k_{i+1}) \ \mapsto \ \sum_{k_i=1}^{r_i} U_i(k_{i-1}, x_i, k_i) U_{i+1}(k_i, x_{i+1}, k_{i+1}). \qquad (2.22)$$

Starting with a TT tensor U with right-orthonormal component functions $U_3, \dots, U_n$, the first micro-iteration in MALS optimises the non-orthonormal contracted component W for $i = 1$, obtaining an optimised tensor $W_{1,2}$, while leaving $U_3, \dots, U_n$ fixed. Afterwards, its unfolding is re-separated approximately, e.g. by a truncated SVD,

$$\big[(\mathbf{W}_{1,2})^{x_2,k_2}_{k_0,x_1}\big] \ \approx \ \big[\widetilde{\mathbf{U}}^{k_1}_{k_0,x_1}\big]\big[\Sigma^{k_1}_{k_1}\big]\big[\widetilde{\mathbf{V}}^{k_2,x_2}_{k_1}\big] \ =: \ \big[(\widetilde{\mathbf{U}}_1)^{k_1}_{k_0,x_1}\big]\big[(\widetilde{\mathbf{U}}_2)^{x_2,k_2}_{k_1}\big], \qquad (2.23)$$

with $\big[(\widetilde{\mathbf{U}}_1)^{k_1}_{k_0,x_1}\big]$ orthogonal and $k_1 = 1, \dots, r_1$ with the truncation rank r_1 chosen such that the error is below a certain threshold, determined in practice for instance by the current residual (Cf. also the part on approximation problems in Sec. 3.1 for another possibility). The resulting left-orthogonal component function $\tilde{U}_1$ is used to update the component function U_1; the procedure then moves on to compute optimal two-component functions $W_{2,3}, W_{3,4}, \dots W_{d-2,d-1}$ for $i = 2, \dots, d-2$ successively in the same way, updating $U_2, \dots, U_{d-1}$ on the way, and then repeats the procedure in the opposite direction, computing $W_{d-1,d}, \dots, W_{2,3}$ and updating $U_d, \dots, U_2$. Although it is in principle possible to discard the part $\hat{U}_{i+1}$ resulting from decomposition (2.23) of $W_{i,i+1}$ because it is optimised in the following step anyway, one can also use $\hat{U}_{i+1}$ and the current component U_{i+2} to construct a reasonable starting guess for the optimisation of the next contracted component $W_{i+1,i+2}$ along the lines of (2.22), as will be done in the algorithm below. For the formulation of the Modified Alternating Linear Scheme (MALS) in terms of retraction operators, we introduce analogously to (2.20) the two-index retraction operator $P_{i,2}$ for $i = i, \dots, d-1$,

$$P_{i,2} \ : \qquad \mathbb{R}^{r_{i-1}\times n_i \times n_{i+1} \times r_{i+1}} \ \to \ \bigotimes_{i=1}^{d} \mathbb{R}^{n_i}, \qquad (2.24)$$

$$P_{i,2}W = [(\mathbf{U}_1)^{k_1}_{k_0}] \cdot .. \cdot [(\mathbf{U}_{i-1})^{k_{i-1}}_{k_{i-2}}][\mathbf{W}^{k_{i+1}}_{k_{i-1}}][(\mathbf{U}_{i+2})^{k_{i+2}}_{k_{i+1}}] \cdot .. \cdot [(\mathbf{U}_d)^{k_d}_{k_{d-1}}] \qquad (2.25)$$

in which

$$[\mathbf{W}^{k_{i+1}}_{k_{i-1}}]: \ \ (x_i, x_{i+1}) \ \mapsto \ [W(k_{i-1}, x_i, x_{i+1}, k_{i+1})]^{r_{i-1}, \ r_{i+1}}_{k_{i-1}=1, k_{i+1}=1} \in \mathbb{R}^{r_{i-1}\times r_{i+1}}$$

is the two-component function defined by W, and the tensor $P_{i,2}W \in \bigotimes_{i=1}^{d} \mathbb{R}^{n_i}$ is built from $U_1, .., U_{i-1}, W, U_{i+2}, .., U_d$ analogeously to (2.12). In terms of $P_{i,2}$ and a given functional $\mathcal{J}$, the MALS algorithm in algorithmic form is now given in Fig. 2.4.

Some general remarks. As we already noted above, convergence analysis for ALS and MALS is not the topic of this work. Here, we only make the following simple observation.

Require: Functional $\mathcal{J} : \mathbb{R}^{n_1 \times \ldots \times n_d} \to \mathbb{R}$,
Starting vector of component functions $(U_1, \ldots, U_d)$.

Right-orthogonalise the components $(U_3, \ldots, U_d)$.

while termination criterion is not fulfiled **do**

for $i = 1, \ldots, d-1$ **do**
(a) Optimisation step: Compute

$$W_{i,i+1} : \quad (k_{i-1}, x_i, x_{i+1}, k_{i+1}) \quad \mapsto \quad W_{i,i+1}(k_{i-1}, x_i, x_{i+1}, k_{i+1})$$

fulfilling

$$W_{i,i+1} = \operatorname{argmin}\{\mathcal{J} \circ P_{i,2} \ : \ \mathbb{R}^{r_{i-1} \times n_i \times n_{i+1} \times r_{i+1}} \to \mathbb{R}.\}$$

(b) Decimation step: Approximate with low rank r_i

$$\big[(\mathbf{W}_{i,i+1})^{x_{i+1},k_{i+1}}_{k_{i-1},x_i}\big] \approx \big[\tilde{\mathbf{U}}^{k_i}_{k_{i-1},x_i}\big]\big[\tilde{\mathbf{V}}^{x_{i+1},k_{i+1}}_{k_i}\big].$$

Update $U_i \leftarrow \tilde{U},\ U_{i+1} \leftarrow \tilde{V}$.
end for
Repeat in reverse order.

end while

Fig. 2.4: The relaxation TT-MALS algorithm.

LEMMA 2.1. *Let $\mathcal{J}$ be a continuously differentiable functional, U be a given tensor and U^+ be the result of one "left to right" step of an ALS or MALS optimisation of the components $U_1, \ldots, U_d$ of U. Then*

$$\mathcal{J}(U^+) \quad < \quad \mathcal{J}(U), \tag{2.26}$$

or U is a stationary point of $\mathcal{J}$.

Lemma 2.1 is proved for $j = 1$ by observing that if in the optimisation of a component function U_i, there exists $\tilde{U}_i$ for which $(\mathcal{J} \circ P_{i,j})(\tilde{U}_i) < (\mathcal{J} \circ P_{i,j})(U_i)$, this implies (2.26). Therefore, if $\mathcal{J}(U^+) = \mathcal{J}(U)$, then $U_1, \ldots, U_d$ are minimisers of the respective component functionals, meaning that all directional derivatives of $\mathcal{J}$ vanish, proving the asserted statement. The analogous argument holds for $j = 2$.
Of course, the ALS and MALS algorithms may converge to a local minimum. Besides this problem, the MALS and ALS method may mainly be hempered by redundancy contained in the multi-linear ansatz. For the manifold $\mathbb{T}^r$ of TT tensors of fixed rank r, this redundancy can be described explicitely by characterizing the tangent space and then be avoided by introducing gauge conditions, see [14]. In a forthcoming publication, we will investigate in detail how the ALS/MALS algorithm are compatible with the tangent space approach and compare it to methods on the full tangent space. Here, we only note that the update $\delta U^{(n),(i)}$ computed in one micro-iteration step of

the ALS algorithm

$$U^{(n),(i+1)} \leftarrow U^{(n),(i)} + \delta U^{(n),(i)},$$
$$\delta U^{(n),(i)} \;=\; \big[(\mathbf{U}_1)^{k_1}\big] \dots \big[(\delta \mathbf{U}_i)^{k_i}_{k_{i-1}}\big] \dots \big[(\mathbf{U}_d)_{k_{d-1}}\big],$$

lies in both the tangent space $\mathcal{T}_U\mathbb{T}^r$ taken at U and in the manifold $\mathbb{T}^r$ itself, while for an updated tensor U^+ resulting from more than one component update as e.g. in Lemma 2.1, $U^+ - U$ is not an element of the tangent space anymore.

3. Applications: Linear equations, eigenvalue problems, and the Fock space approach of electronic structure calculation. In this section, we are concerned with the properties of the TT approach, applied to some optimisation tasks of special interest. In Sec. 3.1, we will investigate the equations obtained from application of the ALS/MALS scheme to SPD linear equations. In Sec. 3.2, we then sketch the modifications needed for the treatment of according eigenvalue problems, and formulate the Fock space ansatz of electronic structure calculation as an example of a constraint optimisation problem that may be treated using the TT format.

3.1. ALS and MALS for linear equations. We turn our attention to the case where the functional $\mathcal{J}$ corresponds to the linear problem (2.5), in which case we have

$$\mathcal{J}(\mathbf{u}) = \frac{1}{2}\langle \mathbf{A}\mathbf{u}, \mathbf{u}\rangle - \langle \mathbf{b}, \mathbf{u}\rangle \tag{3.1}$$

with symmetric positive definite system matrix $\mathbf{A} : \mathbb{R}^{n_1\times\dots\times n_d} \to \mathbb{R}^{n_1\times\dots\times n_d}$ and a right hand side $\mathbf{b} \in \mathbb{R}^{n_1\times\cdots\times n_d}$, defined by the according tensor equation (2.5) introduced in Section 2.1. We will see that the equations resulting from the optimisation of a single component function in a micro-iteration step are also linear equations, inheriting basic properties of the global problems (3.1), see Th. 3.1. Note that for the treatment in this section, the format in which the tensor quantities $\mathbf{A}, \mathbf{b}$ are given is not important.

Micro-iteration equations. For the linear problem, the ALS or MALS approach leads by definition of the algorithms to a sequence of lower-dimensional optimisation tasks to be treated in the micro-iteration steps for the single components U_i, $i = 1, \dots, d$. The according functionals $\mathcal{J} \circ P_{i,j}$ are in this case given by

$$(\mathcal{J} \circ P_{i,j})(V) \;=\; \frac{1}{2}\langle \mathbf{A}\mathbf{P}_{i,j}\mathbf{v}, \mathbf{P}_{i,j}\mathbf{v}\rangle - \langle \mathbf{P}_{i,j}\mathbf{v}, \mathbf{b}\rangle \;, \quad j = 1, 2, \tag{3.2}$$

where $\mathbf{P}_{i,j}\mathbf{v}$ denotes the vectorisation of the tensor $P_{i,j}(V)$. Denoting the vectorisation operation by **vec**, the expression $\mathbf{P}_{i,j}$ in the above can be viewed as a linear operatoracting on the vectorisation of a one- resp. two-component function V by

$$\mathbf{P}_{i,j} : \;\; \mathbb{R}^m \to \mathbb{R}^{n_1\cdot\dots\cdot n_d}, \quad \mathbf{P}_{i,j}\mathbf{v} = \mathbf{vec}\Big(P_{i,j}\big(\mathbf{vec}^{-1}(\mathbf{v})\big)\Big), \tag{3.3}$$

where $m = r_{i-1}n_ir_i$ for $j = 1$, $m = r_{i-1}n_in_{i+1}r_{i+1}$ for $j = 2$. We shall derive a useful more explicit expression for $\mathbf{P}_{i,j}$ below.
In the equation (3.2), we optimise $V \in \mathbb{R}^{r_{i-1}\times n_i\times r_i}$ in the case $j = 1$ corresponding to the ALS and $V \in \mathbb{R}^{r_{i-1}\times n_i\times n_{i+1}\times r_{i+1}}$ for $j = 2$ giving the MALS; $P_{i,j} = P_{i,j,U}$ is

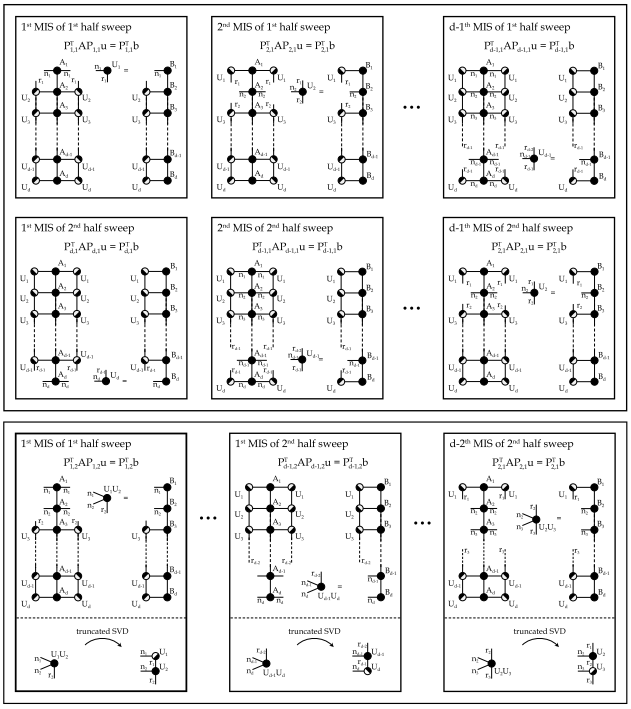


Fig. 3.1: Some of the micro-iteration steps constituting the ALS (a) and MALS (b) scheme for A given in TT format.

defined by the current iterate U. At stationary points V_i of the functional $\mathcal{J} \circ P_{i,j}$, there holds the first-order condition

$$\mathbf{A}_i \mathbf{v}_i - \mathbf{b}_i \;:=\; \mathbf{P}_{i,j}^T \mathbf{A} \mathbf{P}_{i,j} \mathbf{v}_i - \mathbf{P}_{i,j}^T \mathbf{b} \;=\; D(\mathcal{J} \circ P_{i,j}) V_i \;=\; \mathbf{0}, \tag{3.4}$$

a linear equation to be solved in each micro-iteration. Note that $\mathbf{A}_i$ and $\mathbf{b}_i$ depend on the current macro-iterate U via $P_{i,j}$. In Fig. 3.1, we depicted the according course of micro-equations during a ALS and MALS iteration for the case that A is given in TT format. In the i-th micro-iteration step of the ALS and the MALS, the component functions $U_1, \dots, U_{i-1}$ are by construction left-orthonormal and $U_{i+1}, \dots, U_d$ (for ALS) resp. $U_{i+2}, \dots, U_d$ (for MALS) are right-orthonormal (cf. (2.14)). In this case, the retraction matrix $\mathbf{P}_{i,j}$ is orthogonal, so that the matrix $\mathbf{A}_i = \mathbf{P}_{i,j}^T \mathbf{A} \mathbf{P}_{i,j}$ directly inherits important properties of the symmetric operator $\mathbf{A}$. In the following theorem, we summarise these consequences; we then prove the previous statements.

THEOREM 3.1. *For A symmetric positive definite, let $\mathbf{A}_i = \mathbf{P}_{i,j}^T \mathbf{A} \mathbf{P}_{i,j}$ denote the system matrix belonging to a micro-iteration step of the ALS/MALS algorithm (where $j = 1$ and $j = 2$, respectively), with the components $U_1, \dots, U_{i-1}$ left-orthonormal and $U_{i+j}, \dots, U_d$ right-orthonormal.*

(a) In each i-th micro-iteration step of the ALS and MALS algorithm, $\mathbf{A}_i$ is symmetric positive definite. In particular, the systems (3.4) have a unique solution $\mathbf{u}$, defining the update $\widehat{U}_i$ for the i-th component function.

(b) Let $\Lambda_{max}, \Lambda_{min}$ the largest und smallest eigenvalues of A and $\lambda_{max}, \lambda_{min}$ the largest und smallest eigenvalues of $\mathbf{A}_i$. There holds

$$\Lambda_{min} \;\leq\; \lambda_{min} \;\leq\; \lambda_{max} \;\leq\; \Lambda_{max}. \tag{3.5}$$

In particular, throughout the iteration, the condition numbers of the matrices $\mathbf{A}_i$ remain uniformly bounded by

$$cond\ \mathbf{A}_i \quad \leq \quad cond\ \mathbf{A}.$$

Remark. Although it is in principle possible to choose the component functions constituting $\mathbf{P}_{i,j}$ as non-orthonormal, we will see below that then $\mathbf{P}_{i,j}$ need no longer be an orthogonal matrix. Statement (a) then still holds, but (b) is no longer valid in general. The eigenvalues of $\mathbf{A}_i$ may attain *any* positive value, and we observed the deterioration of the condition numbers of the systems (3.4) in practice. Therefore, we strongly advise the reader to maintain left- resp. right-orthonormality of the component functions during the iterations. However, the orthogonality may in principle be replaced by any operator-induced orthogonality such that

$$\mathbf{P}_{i,1}^T \mathbf{A} \mathbf{P}_{i,1} = \mathbf{I}_{\mathbb{R}^{r_{i-1} n_i r_i}} \quad \text{resp.} \quad \mathbf{P}_{i,2}^T \mathbf{A} \mathbf{P}_{i,2} = \mathbf{I}_{\mathbb{R}^{r_{i-1} n_i n_{i+1} r_{i+1}}}.$$

Matrix form of the retraction operator $\mathbf{P}_{i,j}$. We now prove the orthogonality of $\mathbf{P}_{i,j}$, using which Theorem 3.1 can then easily be shown. To do so, we fix U and $i \in \{1, \dots, d+1-j\}$, and give an explicit expression for the operator $\mathbf{P}_{i,j}$ defined by (3.3): We define for $i \in \{2, \dots, d\}$ the i-th left part of U by

$$L_i \in \mathbb{R}^{n_1 \times .. \times n_{i-1} \times r_{i-1}}, \quad L_i(x_1, .., x_{i-1}, k_{i-1}) := \big[\mathbf{U}^{k_1}(x_1)\big] \cdot .. \cdot \big[\mathbf{U}_{k_{i-2}}^{k_{i-1}}(x_{i-1})\big];$$

and by $L_i = [1]$ for $i = 1$. The according right part we define by

$$R_i \in \mathbb{R}^{r_i \times n_{i+1} \times .. \times n_d}, \quad R_i(k_i, x_{i+1}, .., x_d) := \big[\mathbf{U}_{k_{i+1}}^{k_i}(x_{i+1})\big] \cdot .. \cdots \big[\mathbf{U}_{k_{d-1}}(x_d)\big].$$

Note that then

$$U = \sum_{k_{i-1}=1}^{r_{i-1}} \sum_{k_i=1}^{r_i} L_i(x_1, .., x_{i-1}, k_{i-1}) U_i(k_{i-1}, x_i, k_i) R_i(k_i, x_{i+1}, .., x_d).$$

Using the unfoldings

$$\mathbf{L}_i := [(\mathbf{L}_i)_{x_1, \dots, x_{i-1}}^{k_{i-1}}], \quad \mathbf{R}_i := [(\mathbf{R}_i)_{x_{i+1}, \dots, x_d}^{k_{i+1}}],$$

$\mathbf{P}_{i,j}$ is now given by

$$\mathbf{P}_{i,1} := \mathbf{L}_i \otimes \mathbf{I}_{n_i} \otimes \mathbf{R}_i \quad \in \quad \mathbb{R}^{(n_1 \cdot \dots \cdot n_d) \times (r_{i-1} n_i r_i)} \tag{3.6}$$

in the case $j = 1$ designed for the ALS, and by

$$\mathbf{P}_{i,2} := \mathbf{L}_i \otimes \mathbf{I}_{n_i} \otimes \mathbf{I}_{n_{i+1}} \mathbf{R}_{i+1} \quad \in \quad \mathbb{R}^{(n_1 \cdot \dots \cdot n_d) \times (r_{i-1} n_i n_{i+1} r_{i+1})}. \tag{3.7}$$

in the case $j = 2$, corresponding to MALS.

LEMMA 3.2. *Let $j = 1, 2$ and $U_1, \dots, U_{i-1}$ be left-orthonormal, $U_{i+j}, \dots, U_d$ right-orthonormal. Then $\mathbf{P}_{i,j}$ is orthogonal, that is, with $m = r_{i-1}n_i r_i$ for $j = 1$, $m = r_{i-1}n_i n_{i+1} r_{i+1}$ for $j = 2$, there holds*

$$\mathbf{P}_{i,j}^T \mathbf{P}_{i,j} = \mathbf{I} \in \mathbb{R}^{m \times m}.$$

Proof. An easy induction based on the left resp. right orthogonality of the components U_i shows

$$\mathbf{L}_i^T \mathbf{L}_i = \mathbf{I} \in \mathbb{R}^{r_{i-1} \times r_{i-1}}, \quad \mathbf{R}_i \mathbf{R}_i^T = \mathbf{I} \in \mathbb{R}^{r_i \times r_i},$$

from which the result follows by the representations (3.6), (3.7) for $\mathbf{P}_{i,j}$.
□

Theorem 3.1 now easily follows: Lemma 3.2 shows that $\mathbf{P}_{i,j}$ is an injection, making $\mathbf{A}_i$ symmetric positive definite and thus proving (a). Part (b) is a consequence of the Rayleigh-Ritz principle, giving with m defined in Lemma 3.2 that

$$\Lambda_{\min} = \min_{\substack{\mathbf{u} \in \mathbb{R}^{n_1 \cdots n_d} \\ u \neq 0}} \frac{\langle \mathbf{A}\mathbf{u}, \mathbf{u} \rangle}{\langle \mathbf{u}, \mathbf{u} \rangle} \quad \leq \quad \min_{\substack{\mathbf{v} \in \mathbb{R}^m \\ v \neq 0}} \frac{\langle \mathbf{A}\mathbf{P}_{i,j}\mathbf{v}, \mathbf{P}_{i,j}\mathbf{v} \rangle}{\langle \mathbf{P}_{i,j}\mathbf{v}, \mathbf{P}_{i,j}\mathbf{v} \rangle} = \lambda_{\min}.$$

An according estimate for $\lambda_{\max}$ completes the proof of Theorem 3.1.

Approximation problems. As a special case, we consider the problem of approximating a given tensor B. Approximation in the ℓ_2-norm is the special case of the linear equation (2.5) with $\mathbf{A}$ identity operator $\mathbf{I}$. In this case, the component equations read

$$\mathbf{A}_i \mathbf{v} - \mathbf{b}_i \ := \ \mathbf{P}_{i,j}^T \mathbf{P}_{i,j} \mathbf{v} - \mathbf{P}_{i,j}^T \mathbf{b} \ = \ \mathbf{0},$$

so that for the components being left resp. right orthonormal, the orthonormality of $\mathbf{P}_{i,j}$ implies the explicit formula

$$\mathbf{v} = \mathbf{P}_{i,j}^T \mathbf{b}$$

for the components of the ℓ_2-best approximation.
In some practical cases, as e.g. for differential equations, approximation of a given tensor B in the ℓ_2-norm may need to be replaced by approximation in terms of some energy norm induced by an operator A. In this case , we obtain the linear system

$$\mathbf{A}_i \mathbf{v} - \mathbf{b}_i \ := \ \mathbf{P}_{i,j}^T \mathbf{A} \mathbf{P}_{i,j} \mathbf{v} - \mathbf{P}_{i,j}^T \mathbf{A} \mathbf{b} \ = \ \mathbf{0}$$

that then can be treated in the above way.
This procedure may also be used as an inner loop in the decimation step of the MALS algorithm, in which the matrix $B \in \mathbb{R}^{(r_{i-1}n_i) \times (n_{i+1}r_{i+1})}$ belonging a two-compontent tensor $W = W_{i,i+1}$ may be approximated in terms of the $\mathbf{A}$-norm by a TT approximation of order 2. In the above, $\mathbf{B}$ is then the left hand side of the approximation task

$$[(\mathbf{W}_{i,i+1})_{k_{i-1},x_i}^{x_{i+1},k_{i+1}}] \ \approx \ \Big[\Big(\sum_{k_i=1}^{r_i} \tilde{U}_i\big((k_{i-1},x_i),k_i\big) \tilde{U}_{i+1}\big(k_i,(x_{i+1},k_{i+1})\big) \Big)_{k_{i-1},x_i}^{x_{i+1},k_{i+1}} \Big].$$

In this, the rank r_i now has to be chosen corresponding to a sufficient accuracy. A suitable r_i may be obtained by a greedy algorithm: Starting with $r_i = 1$ gives a TT rank-1-approximation $\mathbf{T}_1$ of $\mathbf{B}$, i.e. a rank-1-matrix $\mathbf{T}_1 = s_1 \mathbf{x}_1 \mathbf{y}_1^T$ with $\mathbf{x}_1, \mathbf{y}_1$ normalised. Subtracting this approximation, i.e. letting $\mathbf{B} \leftarrow \mathbf{B} - \mathbf{T}_1$, the new tensor $\mathbf{B}$ can now again be approximated by rank 1 etc., until a desired accuracy is reached. The algorithm provides linearly independent components that can be comprised to a matrix $\mathbf{T} = \sum_{i=1}^{r_1} \mathbf{T}_i$. Let us note that because $d' = 2$, $n'_1 = r_{i-1} n_i$, $n'_2 = n_{i+1} r_{i+1}$, this is a rather small problem compared to the original one. Since $d = 2$, further matrix decomposition techniques (e.g. [18]) also provide other alternatives.

3.2. Eigenvalue equations and the Fock space ansatz. The eigenvalue problem can be treated in a similar fashion as the above linear problems. We start with some general observations and then turn to the example of the Fock space approach for electronic structure calculation.

Eigenvalue equations. To compute the smallest resp. largest eigenvalue of a symmetric matrix $\mathbf{A}$, we can minimise resp. maximise the functional

$$\mathcal{J}(\mathbf{u}) = \frac{1}{2} \frac{\langle \mathbf{A}\mathbf{u}, \mathbf{u} \rangle}{\langle \mathbf{u}, \mathbf{u} \rangle}, \qquad \mathbf{u} \neq 0. \tag{3.8}$$

A short computation using the orthogonality of $\mathbf{P}_{i,j}$ yields the first-order condition

$$\begin{aligned} D(\mathcal{J} \circ P_{i,j})(\mathbf{U}) &= \frac{1}{\|\mathbf{u}\|} \Big(\mathbf{P}_{i,j}^T \mathbf{A} \mathbf{P}_{i,j} \mathbf{u} - \frac{\langle \mathbf{P}_{i,j}^T \mathbf{A} \mathbf{P}_{i,j} \mathbf{u}, \mathbf{u} \rangle}{\langle \mathbf{u}, \mathbf{u} \rangle} \mathbf{P}_{i,j}^T \mathbf{P}_{i,j} \mathbf{x} \Big) \\ &= \frac{1}{\|\mathbf{u}\|} \Big(\mathbf{A}_i \mathbf{u} - \lambda_{\mathbf{u}} \mathbf{u} \Big), \end{aligned}$$

where $\lambda_{\mathbf{u}}$ is the Rayleigh quotient of $\mathbf{u}$ with respect to the matrix $\mathbf{A}_i$. Note that the last line holds only if $\mathbf{P}_{i,j}$ is orthogonal (as is the case for (M)ALS), otherwise the component equations are generalised eigenproblems. If $\mathbf{P}_{i,j}$ is isometric as in the ALS/MALS presented here, a normalised solution from the mircoiteration provides a normalised solution for the original problem.

Electronic Schrödinger equation – general remarks. An example of such a high-dimensional eigenequation is the *electronic Schrödinger equation* [13], the basic equation for a quantitative description of atomic and molecular physics, given by

$$H\Psi \;:=\; \Big(-\frac{1}{2} \sum_{i=1}^{N} \Delta_i - \sum_{i=1}^{N} \sum_{\nu=1}^{K} \frac{Z_\nu}{|x_i - a_\nu|} + \sum_{i<j}^{N} \frac{1}{|x_i - x_j|} \Big) \Psi \;=\; E\Psi. \tag{3.9}$$

It describes the stationary non-relativistic behavior of N electrons exposed to an external field induced by fixed nuclei. A quantity of major interest is the lowest eigenvalue E of the Hamilton operator H, giving the ground state energy of this system. H acts on antisymmetric state functions

$$\Psi \;\in\; \widehat{H}^1_N \;:=\; H^1\big((\mathbb{R}^3 \otimes \{\pm\frac{1}{2}\})^N, \mathbb{C}\big) \;\cap\; \wedge_{i=1}^N L_2(\mathbb{R}^3 \otimes \{\pm\frac{1}{2}\}, \mathbb{C}),$$

where the antisymmetry constraint $\Psi \in \wedge_{i=1}^{N} L_2(\mathbb{R}^3 \otimes \{\pm\frac{1}{2}\}, \mathbb{C})$ reflects the *Pauli principle*, demanding that wave functions of identical fermions, in particular of electrons, have to be antisymmetric with respect to simultaneous permutation of two distinct position and spin variables. The HT and TT format are sensitive to permutation of variables; a naive application of the above TT ansatz for the approximation of Ψ is therefore not appropriate.
We will pursue an alternative strategy, using a binary encoding [13, 36] of the discrete Fock space $\mathcal{F}$ that is standardly used in the formalism of Second Quantisation. Using this approach, we will give a to our minds clearer derivation of the successful DMRG method [3, 40, 41] that then corresponds to the above MALS method applied to this setting. For sake of brevity, we will not consider spin systems explicitly, but refer the reader to the survey papers [34, 12] for deeper insight to the DMRG method and to [44] for a short, comprehensive introduction. We also highlight the recent state-of-the-art review [15] concerned with the numerical treatment of spin systems by ALS and also establishing the connection to the recent developments in tensor product approximation.

Fock space approach. To develop the Fock space formalism announced above, we use antisymmetric tensor products of functions φ_i depending on a single particle variable $\mathbf{x} = (\mathbf{r}, s) \in (\mathbb{R}^3 \times \pm\{\frac{1}{2}\})$, taken from a d-dimensional, L_2-orthonormal discrete one particle basis set

$$\mathcal{X}_d \;:=\; \mathrm{span}\{\varphi_i : i = 1, \dots, d\} \quad \subset \quad H^1(\mathbb{R}^3 \times \{\pm\frac{1}{2}\}), \tag{3.10}$$

where $d \geq N$ is fixed from now on. Every ordered selection $\nu_1, \dots, \nu_M$ of $M \leq d$ indices then gives an M-particle Slater determinant

$$\Psi_{SL}[\nu_1, \dots, \nu_M](\mathbf{r}_1, s_1; \dots; \mathbf{r}_N, s_N) = \frac{1}{\sqrt{M!}} \det(\varphi_{\nu_i}(\mathbf{r}_j, s_j))_{i,j=1}^{M}. \tag{3.11}$$

The ensemble of all such determinants with particle number M forms an orthonormal basis of an antisymmetric discrete M-particle tensor space

$$\mathcal{V}_M^d := \mathrm{span}\{\Psi_{SL}[\nu_1, \dots, \nu_M] \mid 1 \leq \nu_1 < \dots < \nu_M \leq d\}.$$

The sought approximation of the N-electron wave function Ψ is an element of the Fock space

$$\mathcal{F} \;=\; \mathcal{F}_d \;=\; \bigoplus_{M=0}^{d} \mathcal{V}_M^d$$

subject to the constraint that it is constructed solely of N-particle Slater determinants (i.e. $\Psi \in \mathcal{V}_M^d$, while $\mathcal{F}$ contains linear combinations of determinants with $0, \dots, d$ particles). To see the deeper reason behind this reformulation of the problem on the Fock space $\mathcal{F}$, we now index each basis function $\Psi_{SL}[\nu_1, \dots, \nu_M]$ from the Slater basis (3.11) by a binary string $\mu = (\mu_1, \dots, \mu_d)$ of length d, in which we let $\mu_i = 1$ if $i \in \{\nu_1, \dots, \nu_M\}$, $\mu_i = 0$ elsewise. With

$$e^0 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad e^1 = \begin{pmatrix} 0 \\ 1 \end{pmatrix},$$

the mapping defined by

$$\iota : \quad \Psi_{SL}[\nu_1, \ldots, \nu_M] \quad \mapsto \quad e^{\mu_1} \otimes \ldots \otimes e^{\mu_d} \ \in \ \mathcal{W} \ := \ \bigotimes_{i=1}^{d} \mathbb{R}^2$$

is a unitary isomorphism between $\mathcal{F}$ and $\mathcal{W}$. To the latter, tensor decomposition techniques like the TT format in combination with the ALS or MALS method may now be applied without having to deal with the antisymmetry constraint explicitly.

Operators in the binary space $\mathcal{W}$. It remains to formulate the action of the Hamiltonian (3.9) and the constraint condition of N particle functions on the binary space $\mathcal{W}$. Using the well-known creation and annihilation operators [13], the discrete Hamiltonian H_d on the Fock space $\mathcal{F}_d$ and the *particle number operator* P are given by

$$H_d = \sum_{p,q=1}^{d} h_q^p a_q^\dagger a_p + \frac{1}{2} \sum_{p,q,r,s=1}^{d} g_{rs}^{pq} a_r^\dagger a_s^\dagger a_p a_q, \qquad P := \sum_p a_p^\dagger a_p,$$

with h_p^q and $g_{r,s}^{p,q}$ denoting the one and two electron integrals

$$h_q^p := \langle \varphi_q, (\frac{-1}{2}\Delta - V_{core})\varphi_p \rangle \ , \ g_{r,s}^{p,q} := \frac{1}{2} \langle \varphi_r(\mathbf{x}, s_1)\varphi_s(\mathbf{y}, s_2), \frac{\varphi_p(\mathbf{x}, s_1)\varphi_q(\mathbf{y}, s_2)}{|\mathbf{x} - \mathbf{y}|} \rangle.$$

The task is now to minimise the Rayleigh quotient (3.8) associated with H_d under the additional constraint that $P\Psi = N\Psi$, i.e. such that Ψ is an eigenfunction of the particle operator with eigenvalue N. Using the isomorphism ι between $\mathcal{F}$ and $\mathcal{W}$, we now formulate this task and the calculus of second quantisation on the binary space $\mathcal{W}$. We define the according binary annihilation operators by $\mathbf{A}_p = \iota \circ a_p \circ \iota^*$, where ι^* denotes the adoint of ι. The according creation operator is then given by $\mathbf{A}_p^T$, and the operators $\mathbf{A}_p, \mathbf{A}_q^T$ satisfy the usual anti-commutator relations [13, 36]. Moreover, using the 2×2-matrices

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

it can be shown (also see e.g. [36], Sec. 7.1) that there holds

$$\mathbf{A}_p := I \otimes \ldots \otimes I \otimes A_{(p)} \otimes S \otimes \ldots \otimes S,$$

where $A_{(p)}$ indicates that A appears on the p-th position in the product. In particular, the use of the matrices S guarantees the correct sign. The discrete (full CI) Schrödinger equation can be cast into the binary variational form of finding $\mathbf{c} \in \mathcal{W}$ such that

$$\mathbf{c} = \operatorname{argmin}\{\langle \mathbf{Hc}, \mathbf{c} \rangle : \langle \mathbf{c}, \mathbf{c} \rangle = 1 \ , \ \mathbf{Pc} = N\mathbf{c}\},$$

with the binary Hamiltonian $\mathbf{H}$ resp. number operator $\mathbf{P}$, given by

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

$$\mathbf{P} = \sum_{p,q=1}^{d} \mathbf{A}_p^T \mathbf{A}_q = \sum_{p=1}^{d} I \otimes \ldots \otimes B_p \otimes \ldots \otimes I, \quad B := A^T A = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}.$$

With the present TT approach, one now has access to the (full CI) approximate wave function Ψ in the given variational framework, and by contracting all appearing indices in appropriate order, gradients, e.g. Hellman-Feynman forces, reduced density matrices etc. can be computed as usual at polynomial cost. Size consistency is given iff the orbitals are ordered in a way corresponding to how the two subsystems are separated.
A rough scaling estimate of the required complexity depends on how the maximal ranks R of the operator $\mathbf{H}$ scale with the size of the basis set d and number of particle N. Recent results [2] from tensor product approximation of the two electron integrals show $\mathcal{O}(d^2)$ behaviour for high precision ($\sim \mu H$) in the canonical format. We conjecture that for high precision, the TT ranks of U and the operator $\mathbf{H}$ also scale as $\mathcal{O}(d^2)$, while for moderate accuracy ($\sim mH$), $\mathcal{O}(d)$ is sufficient. Assuming $N \sim d$, we obtain for ALS with $\mathbf{H}$ given in canonical format a total scaling of $\mathcal{O}(r^2N^3)$ for the storage and $\mathcal{O}(r^3N^3)$ in computing time if intermediate matrices are precomputed, and $\mathcal{O}(r^2N^2)$ for the storage and $\mathcal{O}(r^3N^4)$ in computing time elsewise, see Section 4.1. In TT format, the rank of $\mathbf{H}$ enters quadraticly; using $\mathbf{H}$ in canonical format therefore is preferable from this point of view.
For the computation of dynamic correlation, the algorithm therefore does not really seem to be not really competitive with CC methods. For static correlation, it is clearly more efficient than most other known methods. Even the density matrix minimisation method [27] in its efficient version using T_1 and T_2 positivity conditions is scaling as $\mathcal{O}(d^6)$ in storage and at least as $\mathcal{O}(d^9)$ in computing time. Since the complexity depends only on the size of the basis set d, the DMRG algorithm might be used together with orbital optimisation in a self-consistent MRSCF cycle [42], see e.g. [25] for a survey. It is likely that the ranks decrease if localised orbitals are used [43]. This may allow the application of several techniques [35] developed for MP2 and CC methods to further improve the scaling behaviour.

4. Computational issues and numerical examples. In this section we turn to practical aspects of the methods derived in Section 3. To treat the linear systems and eigenvalue problems discussed there, we will in the following assume that the operator A defined by (2.3) is given in TT format or in canonical format (which corresponds to a sum of rank-1 TT decompositions), and that the right hand side B is in TT format. If this is not the case, A and B can be decomposed beforehand by the approximation methods from Section 3.1. We first make some general remarks on the quantities that have to be computed during the iteration and on the complexity of the involved quantities and operations; we then give some concrete numerical examples.

4.1. Update of component matrices and complexity. For the treatment of the equations (2.5) and (2.6), the i-th micro-iteration of a sweep requires the application of the matrix $\mathbf{A}_i := \mathbf{P}_{i,j}^T \mathbf{A} \mathbf{P}_{i,j}$ and computation of the quantity $\mathbf{P}_{i,j}^T \mathbf{b}$ (where $\mathbf{b} = \mathbf{u}$ in the eigenvalue case).

Concerning $\mathbf{A}_i$, one can split the TT tensor in a left half $\mathbf{G}_i$ and a right half $\mathbf{H}_{i,j}$ as depicted in Fig. 4.1 for ALS and in Fig. 4.2 for MALS. Given a TT tensor U, the quantities $\mathbf{H}_{d,j}, \mathbf{H}_{d-1,j}, \ldots, \mathbf{H}_{j+1,j} (j = 1, 2)$ can be precomputed and stored if enough storage is available.

The update step $\mathbf{H}_{i,2} \to \mathbf{H}_{i-1,2}$ (for the MALS case) is depicted in Fig. 4.3. For the i-th micro-iteration, one then analogously updates $\mathbf{G}_{i-1}$ to $\mathbf{G}_i$ in the same vein, starting with $\mathbf{G}_1 = \mathbf{A}_1$. One can now apply $\mathbf{A}_i$ to a vector $\mathbf{v}_i$ by contracting

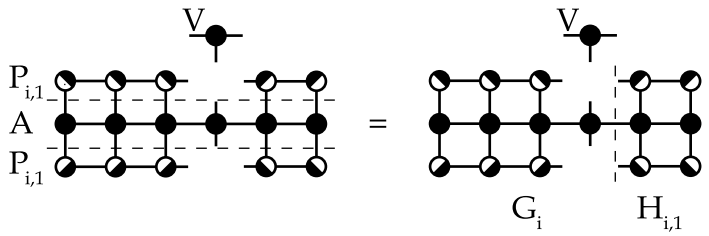


Fig. 4.1: The matrices $\mathbf{G}_i$ and $\mathbf{H}_{i,1}$ used for the application of $\mathbf{A}_i$ in the ALS algorithm, and their application to a one-component iterate V.

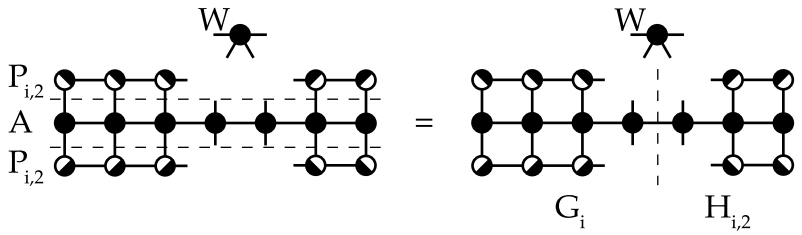


Fig. 4.2: The matrices $\mathbf{G}_i$ and $\mathbf{H}_{i,2}$ used for the application of $\mathbf{A}_i$ in the MALS algorithm, and their application to a two-component iterate W.

the common indices of the corresponding component function V_i with $\mathbf{G}_i$, and then with $\mathbf{H}_i$, see also Fig. 4.1, 4.2. To discuss the complexity of this iteration, we denote by r the maximum of the ranks r_i of U and of B for the linear equation, and by R, depending on how A is given, the maximal rank of the TT decomposition of A or the canonical rank of A. Storage of U, B is then of $\mathcal{O}(r^2nd)$; for A, we have $\mathcal{O}(R^2n^2d)$ in TT format and $\mathcal{O}(Rn^2d)$ in canonical format, respectively. Storage of a matrix $\mathbf{G}_i$ resp. $\mathbf{H}_i$ is of $\mathcal{O}(r^2Rn^2)$. Thus, the total storage requirement is $\mathcal{O}(dr^2R^2n^2)$.

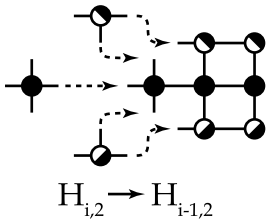


Fig. 4.3: The update step for the right part $\mathbf{H}_{i,2}$ of $\mathbf{A}_i$ in the MALS algorithm.

The computational steps to be taken per sweep consist in the computation of $\mathbf{G}_i$ and $\mathbf{H}_{i,j}$, $1 \le i \le d-j+1$, (scaling as $\mathcal{O}(r^3R^2n^2)$ for TT format and as $\mathcal{O}(r^3Rn^2)$ in the canoncial format) and the computation of the matrices $\mathbf{P}_{i,j}\mathbf{b}$, being of $\mathcal{O}(r^3nd)$ on the whole. Additionally, in each micro-iteration step, application of $\mathbf{G}_i$ and $\mathbf{H}_{i,j}$ to $\mathbf{u}$ has to be performed to obtain $\mathbf{A}_i\mathbf{u}$; here, we obtain $\mathcal{O}(r^3R^2n^2)$ for the ALS-TT step, $\mathcal{O}(r^3Rn^2)$ for ALS with canonical operator, $\mathcal{O}(r^3R^2n^3)$ for MALS with TT operator and $\mathcal{O}(r^3Rn^3)$ for MALS with canonical A. Multiplying these quantities by d for ALS and by $d-1$ for MALS, one sees these are the dominant steps in each sweep, thus giving an estimate for the workload for one sweep of ALS/MALS. Note that the estimates are for the case that the matrices $A_i(k_{i-1}, \cdot, \cdot, k_i)$ are dense; in the sparse case, the n^2-behaviour can be reduced to be linear in n. If n is small or moderate, as

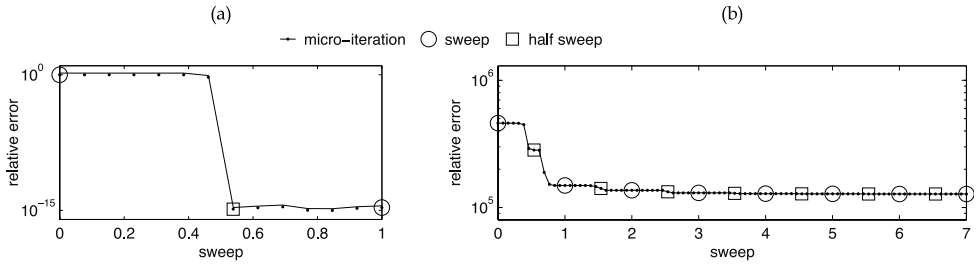


Fig. 4.4: ALS approximation of the one-dimensional Laplacian with TT tensors of constant rank $r_i = 3$ (a) and $r_i = 2$ (b).

is the case in our Fock space setting ($n = 2$ and $n = 4$), the dependence on n is not of importance.

4.2. Numerical examples. In this section, we present some sample numerical experiments using the ALS and MALS algorithm, displaying how the algorithms behave in generic situations. All results presented here were computed on an AMD 2.6 GHz with 4 cores and 32 GB RAM with MATLAB 7.9, using built-in functions and some routines from the TT-Toolbox [30].

Approximation with ALS. Our first example, a "reconstruction test", consisted in the approximation of tensors U of known TT rank by TT tensors of that same rank or higher ranks, using the ALS as described in Section 3.1. In all experiments, a single half sweep of ALS returned a tensor that was a numerically exact approximation of U. Figure 4.4 displays a sample calculation. In this, the TT tensor U to be approximated by ALS was constructed by using the quantised TT approach [17] to turn a tridiagonal matrix of the size $2^p \times 2^p$ (corresponding to a finite difference approximation of the second derivative) to $U \in \mathbb{R}^{(4^p)}$ of order p and of TT ranks $r_i = 3$.

A peculiar detail we observed is that during the first micro-iteration steps, the errors do not decrease; nearly all error improvement in a step is gained in the last step of the half sweep. This effect also occurs in the situation where the approximation ranks are chosen lower than that of the original tensors. Because in this case, more than a half sweep is needed to calculate sufficient precision, this typically results in plateaus in the according convergence curves as displayed in Figure 4.4(b). In this example, approximation of a tensor of order $d = 7$ took 4 sweeps; our observation is that convergence can sometimes take much longer for tensors of higher order d.

As in the case of SVD for the matrix case, a TT approximation does not need to be unique, even in its re-matrification. Figure 4.5 shows another example from our "quantised second derivative" test series (see above), where the approximation computed depends on the inital guess. While the approximations have the same distance from the tensor U, they are distinct, and we conjecture that there may be multiple best approximations if TT decompositions corresponding to multiple singular values are computed.

Approximation with MALS. The big advantage of the MALS algorithm is its feature of automatic rank adaptation. Our experience is that starting with a

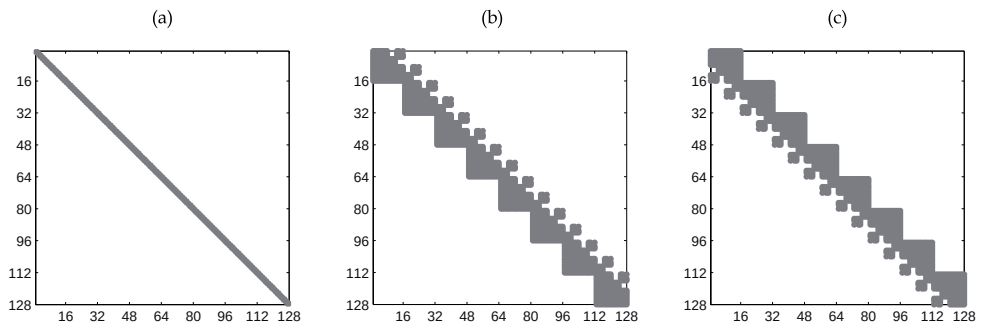


Fig. 4.5: TT Rank-2 ALS approximations of the discrete second derivative: (a) original matrix, (b), (c) two different approximations, obtained from different starting values.

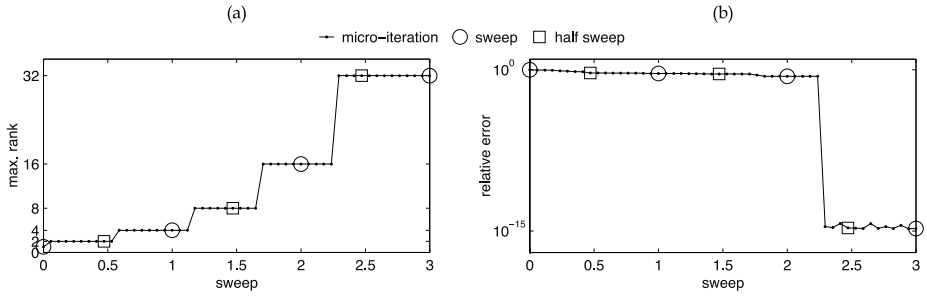


Fig. 4.6: Development of maximum rank (a) and relative error (b) during an approximation with MALS.

rank-1 initial guess will always return a tensor with the rank of the approximated tensor. Figure 4.6 shows the relative error and the maximum ranks during a MALS approximation of a random tensor U of order $d = 10$ with $n_i = 2$ for all $i = 1, \dots, 10$, resulting in a TT rank vector $\underline{r} = (2, 4, 8, 16, 32, 16, 8, 4, 2)$. In this example, it takes five MALS half sweeps to adapt the rank. When starting with an initial guess with the rank of U, MALS always returned U numerically exact after one half sweep.

Solving a linear system with MALS. We tested the behaviour of the MALS algorithm for the solution of linear systems as described in Section 3.1. We started by utilising a *cg* method using TT arithmetic as inner solver with the according parts $W_{i,i+1}$ contracted from the current iterate as in (2.22) taken as initial guess. In this proceeding, we were faced with the problem that in the step optimising the compound function $W_{i,i+1}$, the rank r_i is not *a priori* bounded, and even if the ranks of the solution are moderate, we experienced that the ranks of the iterates may grow to unmanagable sizes during the solution process, only to decrease again when getting near to the solution; see Figure 4.7 for such an example.

To avoid this situation, we therefore utilised a modified MALS procedure: We did not iterate the *cg* method until convergence, but only did some *cg* steps and

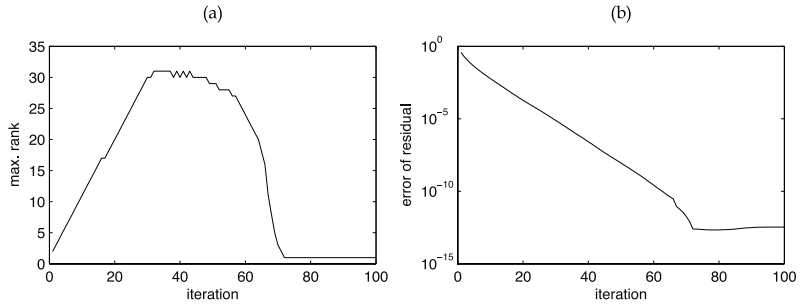


Fig. 4.7: Solution of a linear system in cg TT arithmetic: Maximum ranks of the iterates (a) and the norm of the error (b).

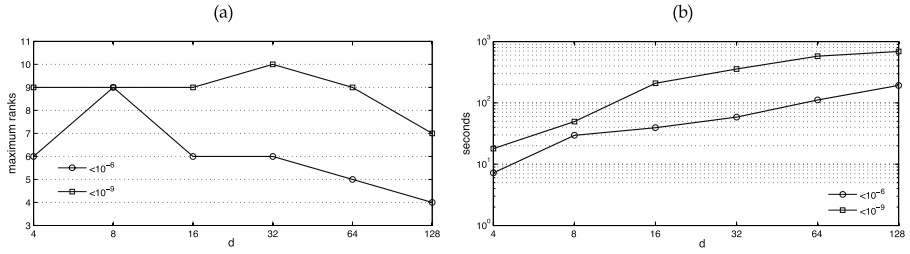


Fig. 4.8: Solution of a linear system with MALS: Maximum ranks of the solutions (a) and the timings (b) with different accuracies.

then computed a low-rank approximation of the iterate, this step then also replacing the SVD decimation step of the MALS procedure. If this approximation yielded no improvement compared with the former iterate, we raised the truncation accuracy to a new level. Although compared to the original MALS, the number of sweeps needed increases by this procedure, the overall cost is reduced due to the lower ranks, thus facilitating the solution of the given linear systems. Another heuristic we found to be useful was to do one CG step with the full TT tensor to get a good initial guess for the MALS algorithm. We note that all these strategies are rather heuristic, and that it would be desirable to have a rigorous strategy preventing the occurrance of high ranks in the approximation of low rank quantities.

In Figure 4.8, we now display the maximum ranks and computation times needed to compute two different accuracies $(10^{-6}, 10^{-9})$ for the solution of a linear system $AU = B$, using as system matrix A a TT approximation of the (canonical format) finite-difference d-dimensional Laplacian ($d \in \{4, 8, 16, 32, 64, 128, 256\}$), discretised on a grid with 10 equally distributed points with Dirchlet boundary conditions. This TT approximation has rank 2, see [28] for a sketch of the proof. As right hand side, we use the discretisation of

$$B(x_1, \ldots, x_d) := \prod_{i=1}^{d} \exp(x_i), \qquad \underline{x} \in [0,1]^d.$$

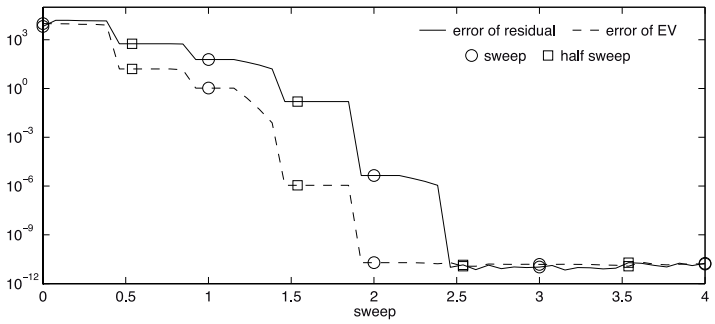


Fig. 4.9: Eigenvector/eigenvalue approximation for a quantised TT operator.

Obviously, this tensor has canonical and TT rank one. The analytical rank of the solution U is unknown; a numerical test yields a maximal TT rank $r_i = 13$. Note that the solution times are (sub-)linear in d, and also that for $d = 8$ and target accuracy of 10^{-6}, MALS "overshoots" this target and returns a solution of higher accuracy, but also of higher rank, which also influences the timing of this iteration. This again stresses the importance of devising rigorous strategies for rank capping.

Solving an eigenvalue problem with ALS. We represented the discretisation of the second order derivate as a quantised TT operator (as in the first experiment) and computed its lowest eigenpair, see Fig. 4.9. The solution has the TT ranks $r_i = 2$, see [32] . The errors may again stagnate during the micro-iterations, resulting in the characteristic plateaus as in the case of linear equations; on the whole though, we observed a fast and stable convergence behaviour.

5. Conclusion and lookout. In this article, we investigated the behaviour and structural properties of the Alternating Linearised Scheme (ALS) and its modification MALS when applied to some common optimisation problems. In order to make use of the structural properties of the underlying optimisation problems, we found that it is important to exploit the fact that the TT tensors can be represented as a left resp. right orthonormalised component functions [14]. The ALS, being particularly simple to implement, shows extraordinarily nice convergence behaviour similar to that of ALS applied to the Tucker format. This behaviour will be complemented by according proofs of convergence in a forthcoming publication. Unfortunately, ALS is sequential by definition. An according Jacobi-like procedure would allow for parallelisation, and both algorithms allow for generalisation to the HT format, for which their convergence properties are still unclear though.
For the MALS, our practical experience concerning stability is not any less positive, although proofs of convergence seem out of reach as long as no rigurous strategy for rank adaption is devised. Optimisation of this step and also the arrangement of indices to which TT is sensitive should be investigated further in the future. In the context of quantum chemistry, the MALS algorithm applied to the binary Fock space corresponds to the successful DMRG algorithm, and we conjecture that in the context of multireference methods, the potential of the MALS/DMRG approach has not fully been exploited and can profit from the connections with the newer developments of

tensor approximation methods in the field of numerical mathematics.

Acknowledgements. This work has been supported by the DFG SPP 1324 "Mathematische Methoden zur Extraktion quantifizierbarer Information aus komplexen Systemen". The authors also thank the DFG research center MATHEON and the Berlin Mathematical School (BMS) for providing travel means used in connection with the development of this work.

REFERENCES

[1] G. Belykin, M. J. Mohlenkamp, *Algorithms for numerical analysis in high dimensions*, SIAM J. Sci. Comp. 26 **6**, p. 2133, 2005.

[2] U. Benedikt, A. Auer, M. Espig, W. Hackbusch, *Tensor Decomposition in post-Hartree Fock Methods*, preprint, available at *http://www.mis.mpg.de/publications/preprints/2010/prepr2010-63.html*, 2010,

[3] G. K.-L. Chan, M. Head-Gordon, *Highly correlated calculations with a polynomial cost algorithm: A study of the density matrix renormalization group*, J. Chem. Phys. 116, p. 4462, 2002.

[4] D. Conte, C. Lubich, *An error analysis of the multi-configuration time-dependent Hartree method of quantum dynamics*, M2AN 44, p. 759, 2010.

[5] M. Espig, *Effziente Bestapproximation mittels Summen von Elementartensoren in hohen Dimensionen*, Ph.D. thesis, 2007.

[6] M. Espig, W. Hackbusch, T. Rohwedder, R. Schneider, *Variational Calculus with Sums of Elementary Tensors of Fixed Rank*, submitted to Num. Math.

[7] C. Eckart, G. Young, *The approximation of one matrix by another of lower rank*, Psychometrika 1 **3**, p. 211, 1936.

[8] A. Falcó, W. Hackbusch, *On Minimal Subspaces in Tensor Representations*, preprint, available at *http://www.mis.mpg.de/publications/preprints/2010/prepr2010-70.html*

[9] L. Grasedyck, *Hierarchical singular value decomposition of tensors*, SIAM. J. Matrix Anal. & Appl. 31, p. 2029, 2010.

[10] E. Hairer, C. Lubich, G. Wanner, *Geometrical Numerical Integration - Structure-Preserving Algorithms for Ordinary Differential Equations*, 2nd edition, Springer, 2006.

[11] W. Hackbusch, S. Kühn, *A new scheme for the tensor representation*, Preprint 2, MPI MIS, 2009.

[12] K. A. Hallberg, *New trends in density matrix renormalization, Advances in Physics*, 55 **5,6** p. 477 526, 2006.

[13] T. Helgaker, P. Jørgensen, J. Olsen, *Molecular electronic-structure theory*, John Wiley & Sons, New York, 2002.

[14] S. Holtz, T. Rohwedder, R. Schneider, *On manifolds of tensors of fixed TT rank*, submitted to Num.Math, DFG SPP 1324 preprint No. 61.

[15] T. Huckle, K. Waldherr, T. Schulte-Herbrüggen, *Computations in Quantum Tensor Networks*, preprint, submitted to Elsevier, 2010.

[16] A. Kapteyn, H. Neudecker, T. Wansbeek, *An approach to n-mode components analysis*, Psychometrika 51, p. 269, 1986.

[17] B. N. Khoromskij, I. V. Oseledets, *DMRG + QTT approach to high-dimensional quantum molecular dynamics*, preprint, available at *http://www.mis.mpg.de/preprints/2010/preprint2010_69.pdf*, 2010.

[18] O. Koch, C. Lubich, *Dynamical Low Rank Approximation*, SIAM J. Matrix Anal. Appl. 29 **2**, p. 434, 2008.

[19] O. Koch, C. Lubich, *Dynamical Low-rank Approximation of Tensors*, SIAM J. Matrix Anal. Appl. 31, p. 2360, 2010.

[20] T. G. Kolda, B. W. Bader, *Tensor decompositions and applications*, SIAM Rev. 51, **3**, 455-500,

[21] P. M. Kroonenberg, J. De Leeuw, *Principal component analysis of three-mode data by means of alternating least squares algorithms*, Psychometrika 45, p. 69, 1980.

[22] C. Lubich, *From Quantum to Classical Molecular Dynamics: Reduced methods and Numerical Analysis*, Zürich Lectures in advanced mathematics, EMS, 2008.

[23] D. Meyer, F. Gatti, G. A. Worth, *The multi-configurational time-dependent Hartree (MCTDH) method*, Physics Reports 324, p. 1, 2000.

[24] D. Meyer, U. Manthe, L. Cederbaum, *The multi-configurational time-dependent Hartree approach*, Chem. Phys. Lett. 165, p. 73, 1990.
[25] K. H. Marti, M. Reiher, *The Density Renormalization Group Algorithm in Quantum Chemistry*, Progress in Physical Chemistry 3, p. 293, 2010.
[26] K.H. Marti, B. Bauer, M. Reiher, M. Troyer, F. Verstraete, *Complete-Graph Tensor Network States: A New Fermionic Wave Function Ansatz for Molecules*, New J.Phys. 12, 103008, 2010.
[27] D. A. Mazziotti, *Reduced density-matrix mechanics with applications to many-electron atoms and molecules*, Wiley-Interscience, John Wiley & Sons, 2007.
[28] I. Oseledets, *Compact matrix form of the d-dimensional tensor decomposition*, submitted to SIAM J. Sci. Comput., 2009.
[29] I. Oseledets, *On a new tensor decomposition*, Doklady Math. 427, no. 2, 2009.
[30] I. Oseledets, *TT Toolbox 1.0: Fast multidimensional array operations in MATLAB*, Preprint 2009-06, INM RAS, August 2009
[31] I.V. Oseledets, E.E. Tyrtyshnikov, *Tensor tree decomposition does not need a tree*, Submitted to Linear Algebra Appl., 2009.
[32] I. Oseledets, E. E. Tyrtyshnikov, Breaking the curse of dimensionality, or how to use SVD in many dimensions, ICM HKBU Research Report 09-03, February 2009 (www.math.hkbu.edu.hk/ICM); SIAM J. Sci. Comput., 2009, to appear.
[33] E. Schmidt, *Zur Theorie der linearen und nichtlinearen Integralgleichungen. I Teil. Entwicklung willkürlichen Funktionen nach System vorgeschriebener*, Math. Ann. 63 , pp. 433, 1907.
[34] U. Schollwöck, *The density-matrix renormalization group*, Rev. Mod. Phys. 77 **1**, p. 259, 2005.
[35] M. Schütz, H.-J. Werner, *Low-order scaling local correlation methods. IV. Linear scaling coupled cluster (LCCSD)*, J. Chem. Phys. 114, p. 661, 2000.
[36] L. A. Takhtajan, *Quantum mechanics for Mathematicians*, AMS Graduate Studies in Mathematics, Vol. 95, 2008.
[37] L. R. Tucker, *Some mathematical notes on three-mode factor analysis*, Psychometrica 31, **3**, p. 279-311, 1966.
[38] C. F. Van Loan, *Tensor network computations in quantum chemistry*, available at www.cs.cornell.edu/cv/OtherPdf/ZeuthenCVL.pdf, 2008.
[39] G. Vidal, *Efficient classical simulation of slightly entagled quantum computation*, Phy. Rev. Letters 91 **14**, 2003.
[40] S. R. White, *Density Matrix Formulation for Quantum Renormalization Groups*, Physical Review Letters 69, p. 2863, 1992.
[41] S. R. White, R. L. Martin, *Ab initio quantum chemistry using the density matrix renormalization group*, J. Chem. Phys. 110, p. 4127, 1999.
[42] T. Yanai, Y. Kurashige, D. Ghosh, G. K.-L. Chan, *Accelerating convergence in iterative solution for large-scale complete active space self-consistent-field calculations*, Int. J. Quantum Chem. 109, p. 2178, 2009.
[43] D. Zgid, M. Nooijen, *Construction of environment states in quantum-chemical density-matrix renormalization group calculations*, J. Chem. Phys. 124, 034103, 2006.
[44] *http://en.wikipedia.org/wiki/Density_matrix_renormalization_group*, last viewed Dec. 6th 2010.

Preprint Series DFG-SPP 1324

http://www.dfg-spp1324.de

Reports

[1] R. Ramlau, G. Teschke, and M. Zhariy. A Compressive Landweber Iteration for Solving Ill-Posed Inverse Problems. Preprint 1, DFG-SPP 1324, September 2008.

[2] G. Plonka. The Easy Path Wavelet Transform: A New Adaptive Wavelet Transform for Sparse Representation of Two-dimensional Data. Preprint 2, DFG-SPP 1324, September 2008.

[3] E. Novak and H. Woźniakowski. Optimal Order of Convergence and (In-) Tractability of Multivariate Approximation of Smooth Functions. Preprint 3, DFG-SPP 1324, October 2008.

[4] M. Espig, L. Grasedyck, and W. Hackbusch. Black Box Low Tensor Rank Approximation Using Fibre-Crosses. Preprint 4, DFG-SPP 1324, October 2008.

[5] T. Bonesky, S. Dahlke, P. Maass, and T. Raasch. Adaptive Wavelet Methods and Sparsity Reconstruction for Inverse Heat Conduction Problems. Preprint 5, DFG-SPP 1324, January 2009.

[6] E. Novak and H. Woźniakowski. Approximation of Infinitely Differentiable Multivariate Functions Is Intractable. Preprint 6, DFG-SPP 1324, January 2009.

[7] J. Ma and G. Plonka. A Review of Curvelets and Recent Applications. Preprint 7, DFG-SPP 1324, February 2009.

[8] L. Denis, D. A. Lorenz, and D. Trede. Greedy Solution of Ill-Posed Problems: Error Bounds and Exact Inversion. Preprint 8, DFG-SPP 1324, April 2009.

[9] U. Friedrich. A Two Parameter Generalization of Lions' Nonoverlapping Domain Decomposition Method for Linear Elliptic PDEs. Preprint 9, DFG-SPP 1324, April 2009.

[10] K. Bredies and D. A. Lorenz. Minimization of Non-smooth, Non-convex Functionals by Iterative Thresholding. Preprint 10, DFG-SPP 1324, April 2009.

[11] K. Bredies and D. A. Lorenz. Regularization with Non-convex Separable Constraints. Preprint 11, DFG-SPP 1324, April 2009.

[12] M. Döhler, S. Kunis, and D. Potts. Nonequispaced Hyperbolic Cross Fast Fourier Transform. Preprint 12, DFG-SPP 1324, April 2009.

[13] C. Bender. Dual Pricing of Multi-Exercise Options under Volume Constraints. Preprint 13, DFG-SPP 1324, April 2009.

[14] T. Müller-Gronbach and K. Ritter. Variable Subspace Sampling and Multi-level Algorithms. Preprint 14, DFG-SPP 1324, May 2009.

[15] G. Plonka, S. Tenorth, and A. Iske. Optimally Sparse Image Representation by the Easy Path Wavelet Transform. Preprint 15, DFG-SPP 1324, May 2009.

[16] S. Dahlke, E. Novak, and W. Sickel. Optimal Approximation of Elliptic Problems by Linear and Nonlinear Mappings IV: Errors in L_2 and Other Norms. Preprint 16, DFG-SPP 1324, June 2009.

[17] B. Jin, T. Khan, P. Maass, and M. Pidcock. Function Spaces and Optimal Currents in Impedance Tomography. Preprint 17, DFG-SPP 1324, June 2009.

[18] G. Plonka and J. Ma. Curvelet-Wavelet Regularized Split Bregman Iteration for Compressed Sensing. Preprint 18, DFG-SPP 1324, June 2009.

[19] G. Teschke and C. Borries. Accelerated Projected Steepest Descent Method for Nonlinear Inverse Problems with Sparsity Constraints. Preprint 19, DFG-SPP 1324, July 2009.

[20] L. Grasedyck. Hierarchical Singular Value Decomposition of Tensors. Preprint 20, DFG-SPP 1324, July 2009.

[21] D. Rudolf. Error Bounds for Computing the Expectation by Markov Chain Monte Carlo. Preprint 21, DFG-SPP 1324, July 2009.

[22] M. Hansen and W. Sickel. Best m-term Approximation and Lizorkin-Triebel Spaces. Preprint 22, DFG-SPP 1324, August 2009.

[23] F.J. Hickernell, T. Müller-Gronbach, B. Niu, and K. Ritter. Multi-level Monte Carlo Algorithms for Infinite-dimensional Integration on $\mathbb{R}^{\mathbb{N}}$. Preprint 23, DFG-SPP 1324, August 2009.

[24] S. Dereich and F. Heidenreich. A Multilevel Monte Carlo Algorithm for Lévy Driven Stochastic Differential Equations. Preprint 24, DFG-SPP 1324, August 2009.

[25] S. Dahlke, M. Fornasier, and T. Raasch. Multilevel Preconditioning for Adaptive Sparse Optimization. Preprint 25, DFG-SPP 1324, August 2009.

[26] S. Dereich. Multilevel Monte Carlo Algorithms for Lévy-driven SDEs with Gaussian Correction. Preprint 26, DFG-SPP 1324, August 2009.

[27] G. Plonka, S. Tenorth, and D. Roşca. A New Hybrid Method for Image Approximation using the Easy Path Wavelet Transform. Preprint 27, DFG-SPP 1324, October 2009.

[28] O. Koch and C. Lubich. Dynamical Low-rank Approximation of Tensors. Preprint 28, DFG-SPP 1324, November 2009.

[29] E. Faou, V. Gradinaru, and C. Lubich. Computing Semi-classical Quantum Dynamics with Hagedorn Wavepackets. Preprint 29, DFG-SPP 1324, November 2009.

[30] D. Conte and C. Lubich. An Error Analysis of the Multi-configuration Time-dependent Hartree Method of Quantum Dynamics. Preprint 30, DFG-SPP 1324, November 2009.

[31] C. E. Powell and E. Ullmann. Preconditioning Stochastic Galerkin Saddle Point Problems. Preprint 31, DFG-SPP 1324, November 2009.

[32] O. G. Ernst and E. Ullmann. Stochastic Galerkin Matrices. Preprint 32, DFG-SPP 1324, November 2009.

[33] F. Lindner and R. L. Schilling. Weak Order for the Discretization of the Stochastic Heat Equation Driven by Impulsive Noise. Preprint 33, DFG-SPP 1324, November 2009.

[34] L. Kämmerer and S. Kunis. On the Stability of the Hyperbolic Cross Discrete Fourier Transform. Preprint 34, DFG-SPP 1324, December 2009.

[35] P. Cerejeiras, M. Ferreira, U. Kähler, and G. Teschke. Inversion of the noisy Radon transform on $SO(3)$ by Gabor frames and sparse recovery principles. Preprint 35, DFG-SPP 1324, January 2010.

[36] T. Jahnke and T. Udrescu. Solving Chemical Master Equations by Adaptive Wavelet Compression. Preprint 36, DFG-SPP 1324, January 2010.

[37] P. Kittipoom, G. Kutyniok, and W.-Q Lim. Irregular Shearlet Frames: Geometry and Approximation Properties. Preprint 37, DFG-SPP 1324, February 2010.

[38] G. Kutyniok and W.-Q Lim. Compactly Supported Shearlets are Optimally Sparse. Preprint 38, DFG-SPP 1324, February 2010.

[39] M. Hansen and W. Sickel. Best m-Term Approximation and Tensor Products of Sobolev and Besov Spaces – the Case of Non-compact Embeddings. Preprint 39, DFG-SPP 1324, March 2010.

[40] B. Niu, F.J. Hickernell, T. Müller-Gronbach, and K. Ritter. Deterministic Multi-level Algorithms for Infinite-dimensional Integration on $\mathbb{R}^{\mathbb{N}}$. Preprint 40, DFG-SPP 1324, March 2010.

[41] P. Kittipoom, G. Kutyniok, and W.-Q Lim. Construction of Compactly Supported Shearlet Frames. Preprint 41, DFG-SPP 1324, March 2010.

[42] C. Bender and J. Steiner. Error Criteria for Numerical Solutions of Backward SDEs. Preprint 42, DFG-SPP 1324, April 2010.

[43] L. Grasedyck. Polynomial Approximation in Hierarchical Tucker Format by Vector-Tensorization. Preprint 43, DFG-SPP 1324, April 2010.

[44] M. Hansen und W. Sickel. Best m-Term Approximation and Sobolev-Besov Spaces of Dominating Mixed Smoothness - the Case of Compact Embeddings. Preprint 44, DFG-SPP 1324, April 2010.

[45] P. Binev, W. Dahmen, and P. Lamby. Fast High-Dimensional Approximation with Sparse Occupancy Trees. Preprint 45, DFG-SPP 1324, May 2010.

[46] J. Ballani and L. Grasedyck. A Projection Method to Solve Linear Systems in Tensor Format. Preprint 46, DFG-SPP 1324, May 2010.

[47] P. Binev, A. Cohen, W. Dahmen, R. DeVore, G. Petrova, and P. Wojtaszczyk. Convergence Rates for Greedy Algorithms in Reduced Basis Methods. Preprint 47, DFG-SPP 1324, May 2010.

[48] S. Kestler and K. Urban. Adaptive Wavelet Methods on Unbounded Domains. Preprint 48, DFG-SPP 1324, June 2010.

[49] H. Yserentant. The Mixed Regularity of Electronic Wave Functions Multiplied by Explicit Correlation Factors. Preprint 49, DFG-SPP 1324, June 2010.

[50] H. Yserentant. On the Complexity of the Electronic Schrödinger Equation. Preprint 50, DFG-SPP 1324, June 2010.

[51] M. Guillemard and A. Iske. Curvature Analysis of Frequency Modulated Manifolds in Dimensionality Reduction. Preprint 51, DFG-SPP 1324, June 2010.

[52] E. Herrholz and G. Teschke. Compressive Sensing Principles and Iterative Sparse Recovery for Inverse and Ill-Posed Problems. Preprint 52, DFG-SPP 1324, July 2010.

[53] L. Kämmerer, S. Kunis, and D. Potts. Interpolation Lattices for Hyperbolic Cross Trigonometric Polynomials. Preprint 53, DFG-SPP 1324, July 2010.

[54] G. Kutyniok and W.-Q Lim. Shearlets on Bounded Domains. Preprint 54, DFG-SPP 1324, July 2010.

[55] A. Zeiser. Wavelet Approximation in Weighted Sobolev Spaces of Mixed Order with Applications to the Electronic Schrödinger Equation. Preprint 55, DFG-SPP 1324, July 2010.

[56] G. Kutyniok, J. Lemvig, and W.-Q Lim. Compactly Supported Shearlets. Preprint 56, DFG-SPP 1324, July 2010.

[57] A. Zeiser. On the Optimality of the Inexact Inverse Iteration Coupled with Adaptive Finite Element Methods. Preprint 57, DFG-SPP 1324, July 2010.

[58] S. Jokar. Sparse Recovery and Kronecker Products. Preprint 58, DFG-SPP 1324, August 2010.

[59] T. Aboiyar, E. H. Georgoulis, and A. Iske. Adaptive ADER Methods Using Kernel-Based Polyharmonic Spline WENO Reconstruction. Preprint 59, DFG-SPP 1324, August 2010.

[60] O. G. Ernst, A. Mugler, H.-J. Starkloff, and E. Ullmann. On the Convergence of Generalized Polynomial Chaos Expansions. Preprint 60, DFG-SPP 1324, August 2010.

[61] S. Holtz, T. Rohwedder, and R. Schneider. On Manifolds of Tensors of Fixed TT-Rank. Preprint 61, DFG-SPP 1324, September 2010.

[62] J. Ballani, L. Grasedyck, and M. Kluge. Black Box Approximation of Tensors in Hierarchical Tucker Format. Preprint 62, DFG-SPP 1324, October 2010.

[63] M. Hansen. On Tensor Products of Quasi-Banach Spaces. Preprint 63, DFG-SPP 1324, October 2010.

[64] S. Dahlke, G. Steidl, and G. Teschke. Shearlet Coorbit Spaces: Compactly Supported Analyzing Shearlets, Traces and Embeddings. Preprint 64, DFG-SPP 1324, October 2010.

[65] W. Hackbusch. Tensorisation of Vectors and their Efficient Convolution. Preprint 65, DFG-SPP 1324, November 2010.

[66] P. A. Cioica, S. Dahlke, S. Kinzel, F. Lindner, T. Raasch, K. Ritter, and R. L. Schilling. Spatial Besov Regularity for Stochastic Partial Differential Equations on Lipschitz Domains. Preprint 66, DFG-SPP 1324, November 2010.

[67] E. Novak and H. Woźniakowski. On the Power of Function Values for the Approximation Problem in Various Settings. Preprint 67, DFG-SPP 1324, November 2010.

[68] A. Hinrichs, E. Novak, and H. Woźniakowski. The Curse of Dimensionality for Monotone and Convex Functions of Many Variables. Preprint 68, DFG-SPP 1324, November 2010.

[69] G. Kutyniok and W.-Q Lim. Image Separation Using Shearlets. Preprint 69, DFG-SPP 1324, November 2010.

[70] B. Jin and P. Maass. An Analysis of Electrical Impedance Tomography with Applications to Tikhonov Regularization. Preprint 70, DFG-SPP 1324, December 2010.

[71] S. Holtz, T. Rohwedder, and R. Schneider. The Alternating Linear Scheme for Tensor Optimisation in the TT Format. Preprint 71, DFG-SPP 1324, December 2010.