

LUDWIG-MAXIMILIANS UNIVERSITÄT MÜNCHEN

A THESIS SUBMITTED FOR THE DEGREE OF

MASTER OF SCIENCE

**A Survey of Low-Rank Methods with
Applications in Computational Physics
and Spectra of Fully-Correlated
Helium**

Author:
Jonas Bucher

Supervisor:
Prof. Dr. Armin Scrinzi

17. August 2018



Abstract

This thesis is divided into three disjoint parts. We start by introducing various tensor decompositions based on different rank notions; these are known as *canonical decomposition*, *higher order SVD*, and *tensor networks*. Their mathematical properties are presented as well as use-cases in real-life computational physics.

The remainder of this text concerns fully-correlated Helium spectra. We first discuss optimizations in the computational implementation of the electron-electron repulsion, which accounts for a large part of computational efforts. Grid-based methods are presented and compared to multipole-based application and coupled angular momenta.

The last part presents spectra of the six-dimensional calculations, made possible by the use of a preponderance rule and similar constraints as well as methods like `tSURFF`. These are of practical interest as there is a not-yet-resolved discrepancy between emission angles measured in experiment [1, 2] and numerical results [3, 4]. We find that systematic inclusion of correlations produces results no different from single active electron calculations.

Acknowledgements

First and foremost I would like to express my gratitude to my supervisor, Prof. Dr. Armin Scrinzi, for all his time and effort he invested in this project. He guided me during this work and was always able to intrigue and encourage me to follow all the ideas that came up. His patience in answering all my questions in great detail is deeply appreciated.

I am very grateful to my parents for all their encouragement and support, without whom I would never have enjoyed so many opportunities.

Contents

1. Tensor Decompositions	1
1.1. Motivation	1
1.2. The Matrix Case	2
1.2.1. Singular Value Decomposition	2
1.2.2. \mathcal{H} Matrices	5
1.2.3. Product Representations	14
1.3. Higher Order Tensors	17
1.3.1. Notation	17
1.3.2. Generalizing Rank	18
1.3.3. Higher Order SVD	20
1.4. Constraining Correlations	25
1.4.1. Distinguishability	26
1.4.2. The Dirac-Frenkel Variational Principle	27
1.4.3. Hartree Type Approximations and CP States	27
1.4.4. Tucker States	31
1.4.5. Matrix Product States	33
1.4.6. Tensor Networks	42
1.5. Summary	47
2. Thoughts on Two Electron Interaction	49
2.1. Motivation	49
2.2. Multipole Expansion	49
2.3. Unconstrained Basis	51
2.3.1. Straight-Forward Scheme	51
2.3.2. Explicit Scheme	52
2.3.3. Tensor Product Scheme	54
2.3.4. Product Grid	54
2.3.5. η Grid	58
2.3.6. Thoughts on Grid Sizes	62
2.3.7. Coupled Angular Momenta	62
2.3.8. Further Ideas	67
2.4. Constrained Basis	67
2.4.1. Explicit Schemes	68
2.4.2. Product Scheme	68
2.4.3. Grid Transformations	68
2.4.4. Coupled Angular Momenta	71
2.5. Conclusion	75

3. Helium in an Ultrashort Laser Pulse	79
3.1. Physical Motivation	79
3.1.1. The Attosecond Clock	79
3.1.2. Classical Picture	79
3.1.3. Experimental and Numerical Results	81
3.2. Technology	83
3.2.1. tSURFF	83
3.2.2. irECS	85
3.2.3. tRecX	86
3.3. Results	90
3.3.1. Observables and Convergence	90
3.3.2. Convergence Parameters	93
3.3.3. The Pulse	94
3.3.4. The Helium Model	94
3.3.5. Potential Cutoff	96
3.3.6. Full Helium	100
3.4. Conclusion	102
A. Some Calculations in Greater Detail	103
A.1. MPS Transfer Matrices are Completely Positive	103
A.2. Estimating Correlations in MPS	104
B. Convergence of Observables	107
B.1. Helium Model	107
B.2. Helium	112
Bibliography	117

List of Figures

1.1. Index tree for $n = 4$	7
1.2. P_2 blocking exemplified for $n = 4$	8
1.3. P'_2 blocking exemplified for $n = 8$	9
1.4. Frobenius norm error of the truncation to $\mathcal{H}k$ format of the kernel $\log x - y $ discretized in a normed, piecewise constant basis, for two different blockings	14
1.5. Matrices U and V for a 4×4 P_2 $\mathcal{H}1$ matrix	15
1.6. Geometrical interpretation of a CP state	30
1.7. Geometrical interpretation of a Tucker state	32
1.8. Quotient of first and second singular value of a product of l random quantum channels	39
1.9. Tensor networks for a scalar α , a vector v , a matrix A , and a tensor H over the vector space \mathbb{C}^n	43
1.10. Some examples on arithmetic in tensor networks	43
1.11. Tensor network representation of a MPS with open boundary conditions	44
1.12. Tensor network for the MPS transfer matrices $E_S^{(f)}$	44
1.13. Comparison of MPS/TT, CP and Tucker representations in terms of tensor networks	45
1.14. Two-dimensional PEPS state with open boundary conditions	46
1.15. \mathcal{H} Tucker format for $d = 8$	47
2.1. Comparison of the schemes $F1$ and $F2$	53
2.2. Apply count for the scheme P	55
2.3. Apply count for the scheme $G1$	58
2.4. Apply count for the scheme $G2$	61
2.5. Quadrature errors ϵ of the associated Legendre matrix elements without selection rules	63
2.6. Quadrature errors ϵ of the associated Legendre matrix elements	64
2.7. Applications count of the schemes $C1$ and $C2$	66
2.8. Applications count of the schemes $F1$ and $F2$ under the constraints $C_U = C_{lm} = 3$	69
2.9. Applications count of the scheme $G2$ for both full and minimal grids under the constraints $C_U = C_{lm} = 3$	70
2.10. Applications count of the schemes $C1$ and $C2$ under the constraints $C_U = 3$ and $l + p - M < 6$	72
2.11. Angular distribution of the probability during time propagation of an example situation	73

2.12. Applications count of the schemes $C1$ and $C2$ under the constraints $C_{ll} = 3$ and $J - M < 3$	74
2.13. Comparison of different application schemes for the full basis L^4	76
2.14. Comparison of different application schemes for the constrained basis	77
3.1. The two ionization channels of a bound electron subjected to a strong Laser field	80
3.2. Visualization of the classical interpretation of ionization	82
3.3. Splitting of the wave-function into bound, singly-ionized and doubly ionized parts employing \mathfrak{tSURFF} for a two-electron system	85
3.4. Spectrum of a Schrödinger Hamiltonian under exterior complex scaling	87
3.5. The index tree at the core of \mathfrak{tRecX}	88
3.6. Vector and index tree structures side-by-side	89
3.7. The modified constraints used in practice	93
3.8. The 400nm pulse used in this section	95
3.9. Converged observables for the Helium model	97
3.10. Spectrum resolved with respect to k and ϕ for $\eta = 0$	98
3.11. Total spectra of the Helium model for various values of potential smoothing and cutoff radius	99
3.12. Converged total spectrum and emission angles of the full Helium cal- culation compared to the Helium model	101
3.13. Channel resolved total spectrum of the full Helium calculation	102
B.1. Convergence of the total spectrum of the Helium model with respect to radial and angular basis	108
B.2. Convergence of the total spectrum of the Helium model with respect to the remaining parameters	110
B.3. Convergence of the total spectrum of Helium, part one	114
B.4. Convergence of the total spectrum of Helium, part two	115

List of Tables

1.1. Comparison of some properties of the discussed tensor decompositions	48
3.1. Emission angles of the Helium model for different potential cutoff and smoothing radii	99
3.2. Converged emission angles for the full Helium calculation compared to the Helium model	100
B.1. Values for convergence of total spectrum to 3% resp. 5%, emission angles to 1° resp. 3° for the Helium model	107
B.2. Convergence of the emission angles of the Helium model with respect to radial and angular basis	109
B.3. Convergence of the emission angles of the Helium model with respect to the remaining parameters	111
B.4. Values for convergence of total spectrum to 5%, emission angles to 1° for the full Helium calculation	112
B.5. Convergence of the emission angles of Helium	113

1. Tensor Decompositions

1.1. Motivation

When doing physics on a computer, the task most critical with respect to performance is often handling discretizations of multidimensional quantities. As an example setting, when doing quantum mechanics on $L^2(\mathbb{R}^d)$ using a basis $\{e_1, \dots, e_n\} \subset L^2(\mathbb{R})$, one needs to discretize, store, and apply the Hamiltonian H in this basis

$$H_{i_1 \dots i_d j_1 \dots j_d} = \langle e_{i_1} \otimes \dots \otimes e_{i_d} | H | e_{j_1} \otimes \dots \otimes e_{j_d} \rangle.$$

To do Schrödinger quantum mechanics, we need to discretize a wave-function $\psi_{i_1 \dots i_d}$ and are left with matrix-vector products $\psi'_{i_1 \dots i_d} = H_{i_1 \dots i_d j_1 \dots j_d} \psi_{j_1 \dots j_d}$ as elementary operations. We can accomplish this if H fits within the available memory and we have enough computation time to apply $H\psi$ as often as necessary.

The tensor $H_{i_1 \dots i_d j_1 \dots j_d}$ consists of n^{2d} complex numbers, applying it requires n^{2d} complex multiplications.¹ This power- d scaling is known as the *curse of dimensions*. To represent a physical situation to a satisfying accuracy, the discretization must be chosen fine enough, i. e. n needs to be large enough. However, the curse of dimensions puts a very strong limit on n given d . In the case $d = 6$, which we will be concerned with mostly, doubling the size of the single-axis discretization n leads to $2^{12} = 4096$ as many elements in the Hamiltonian.

So in the end, to describe a physical process, we need to discretize a state in configuration space (scaling with power- d), and we need knowledge about the interaction of all parts of configuration space with all others (an additional square in scaling).

The configuration space can often be constrained to only allow for regions that contribute to the process under investigation. As an example, if the potential has a certain symmetry and we know the value of the associated conserved charge at initial time, we may restrict the discretization to that value of the charge. So, by choosing a well-thought-out discretization of $L^2(\mathbb{R}^d)$, one is able to limit the configuration space to parts that are important.

In a high-dimensional situation one is often able to reproduce physical results well by limiting the entanglement in the basis. I. e. a discretization of $L^2(\mathbb{R}^d)$ in terms of $\mathbb{C}^n \otimes \dots \otimes \mathbb{C}^n$ allows for arbitrary correlations; however, it is often reasonable to choose a product state basis $\mathbb{C}^n \oplus \dots \oplus \mathbb{C}^n$, which does not allow for any correlations (Hartree). Methods like MCTDH [5, 6] or Matrix Product States [7–9], limiting the

¹We will always assume that an addition is cost-free and only be concerned with the number of multiplications. The justification for this simplification is that a multiplication is usually followed by an addition, so the two numbers will scale equally.

amount of correlation possible, are set in between. As opposed to the first example, these constraints are of a non-linear nature and take on the curse of dimensions.

The fact that we need to know how any part of the configuration space interacts with any other part remains. However, in practice we do not expect the interaction of any part with any other to be “of equal importance,” or equally “complicated.” Thinking of the repulsive interaction between two electrons, in regions where they are well separated this interaction is on the one hand less strong and on the other hand does not vary much with the electron positions; it is less complicated.

If we are able to exploit this and store and apply a tensor using significantly less numbers or multiplications than naively expected, we say, the tensor is stored in a *data-sparse* manner.

How to make such intuitions more clear is the purpose of this part. We will start by surveying methods on matrices, where the key is to represent a wide class of different matrices within one scheme that requires less than N^2 numbers to store an $N \times N$ matrix. To this end, after recalling the *singular value decomposition*, we will introduce *hierarchical matrices* and discuss examples, where these yield data-sparse representation. In an effort to generalize SVD and hierarchical matrices, we introduce *product representations* and give a few insights into approximations.

Generalizing the concept “rank” to tensors of higher order proves to be mathematically interesting; we will discuss various concepts such as *canonical decomposition* and *higher order SVD* in the following section. As we will see, these concepts are most useful if not applied to operators or interactions but to states. We find the interpretation of these “low-rank states” to be a constraining of correlations.

1.2. The Matrix Case

In this part, we will limit ourselves to matrices. These might arise as discretizations of higher-dimensional quantities, i.e. be unfoldings of tensors (see Section 1.3.1), but we only think about matrices. The first data-sparse representation of a matrix beyond a sparse matrix that comes to mind is the *singular value decomposition*.

1.2.1. Singular Value Decomposition

Any matrix $A \in \mathbb{C}^{m \times n}$ can be decomposed in terms of its singular values

$$A_{ij} = \sum_{\alpha=1}^k \sigma_{\alpha} U_{i\alpha} V_{j\alpha}^* = U \sigma V^{\dagger}, \quad U \in \mathbb{C}^{m \times k}, \quad \sigma \in \mathbb{R}^{k \times k}, \quad V \in \mathbb{C}^{n \times k}, \quad (1.1)$$

where σ is diagonal with positive entries $\sigma_i > 0$ and U and V are unitary matrices.² k denotes the rank of the matrix. We will call such a representation a *singular value*

²A matrix $A \in \mathbb{C}^{n \times n}$ is called unitary, if $U^{\dagger}U = UU^{\dagger} = \text{Id}$. We call a non-square matrix $A \in \mathbb{C}^{n \times k}$ with $k < n$, (semi-)unitary if $U^{\dagger}U = \text{Id}$; often we drop the “semi,” as a non-square matrix cannot be unitary.

decomposition of A . One quickly verifies that any decomposition of A in this manner satisfies:

$$\begin{aligned} (AA^\dagger)_{ij} U_{j\alpha} &= \sigma_\alpha^2 U_{i\alpha}, \\ (A^\dagger A)_{ij} V_{j\alpha} &= \sigma_\alpha^2 V_{i\alpha}, \end{aligned}$$

i. e.

1. the squares of the entries of the diagonal matrix σ are the eigenvalues of AA^\dagger and $A^\dagger A$,
2. the columns of U form an orthonormal eigenbasis of AA^\dagger called *left-singular vectors*,
3. the columns of V form an orthonormal eigenbasis of $A^\dagger A$ called *right-singular vectors*.

Uniqueness of the SVD These properties imply that any two SVDs of A are equivalent up to

1. reordering the singular values and their corresponding vectors in U and V ,
2. a unitary transformation on every degenerate subspace (In the case $i \neq j \Rightarrow \sigma_i \neq \sigma_j$ this means a phase applied to every column in U and V).

Let us make this mathematically precise. Let

$$A = U\sigma V^\dagger = U'\sigma'V'^\dagger$$

be two SVDs of the same matrix. Then, as both σ and σ' consist of positive eigenvalues of the same matrix, we can write $\sigma' = P^T \sigma P$, where P is a representation of a permutation $\pi : \{1, \dots, n\} \rightarrow \{1, \dots, n\}$

$$P_{ij} = \delta_{i\pi(j)}, \quad P_{ij}^T = P_{ij}^{-1} = \delta_{\pi(i)j}.$$

P implements the reordering mentioned in the first point. As to the second point, $U'P^{-1}$ satisfies the same eigenvalue equation as U with the same eigenvalues σ^2 ; that implies

$$U'P^{-1} = UT, \quad \text{where } [\sigma^2, T] = 0.$$

In case all σ_i are different, this equation can only be satisfied by $T \in U(1) \oplus \dots \oplus U(1)$, a diagonal matrix of phases.³ If the σ_i are degenerate, we have, for example, $T \in$

³By $U(n)$ we denote the group of unitary $n \times n$ matrices.

$U(3) \oplus U(1) \oplus \dots \oplus U(1)$. We call T a unitary transformation on every degenerate subspace and arrive at

$$U' = UTP, \quad V' = VTP, \quad \sigma' = P^T \sigma P.$$

Owed to these strong uniqueness properties, we will call any SVD *the* SVD of A .

Low-Rank Approximations We can approximate any matrix $A = U\sigma V^\dagger$ by a rank- k matrix A_k by setting all but the highest k singular values in σ to zero. We assume the singular values to be ordered by decreasing magnitude; then

$$(A_k)_{ij} = \sum_{\alpha=1}^k \sigma_\alpha U_{i\alpha} V_{j\alpha}^*.$$

This approximation satisfies the optimality condition under the Frobenius norm $\|A\|_F = \sqrt{\text{tr}[A^\dagger A]}$

$$\|A - A_k\|_F = \min_{\text{rank}(\tilde{A}) \leq k} \|A - \tilde{A}\|_F, \quad (1.2)$$

which is known as the *Eckart-Young* theorem [10]. In particular, there is an optimal rank- k approximation of any matrix A . This approximation is unique up to degenerate singular values; i. e. if there is no freedom in choosing the highest k singular values, the rank- k approximation of a matrix is unique. We easily verify

$$\|A - A_k\|_F^2 = \sum_{\alpha=k+1}^{\text{rank}(A)} \sigma_\alpha^2. \quad (1.3)$$

This means, detection of low-rank matrices is a straight-forward task. One finds the highest few eigenvalues of AA^\dagger . If these decay sufficiently fast below a certain threshold, the matrix can be considered low-rank and represented and applied using the singular vectors.

In the setting Schrödinger quantum mechanics one is left with self-adjoint matrices and an SVD is nothing but a diagonalization of the matrix with singular values equal up to a sign to eigenvalues. In general, a Schrödinger type Hamiltonian $H = -\Delta + V$ has some, possibly few, bound states and a large continuum of eigenstates with positive eigenvalues. This means, H will not be of low rank. Even if we restrict our viewpoint to the potential, as a multiplication operator it will in realistic cases not be of low rank. This is easy to see as we can always transform to a grid basis by diagonalizing the position operator. If, for example, we are using a polynomial basis, this grid consists of the quadrature points (DVR) and the potential can be approximated well by a low-rank matrix if it is close to zero at most quadrature points.

Oversampling a Potential Let us look at an example where a low-rank structure arises naturally. Suppose we have $A \in L^2(\mathbb{R}^2) \cong L^2(\mathbb{R}) \otimes L^2(\mathbb{R})$ and $P_k : L^2(\mathbb{R}) \rightarrow L^2(\mathbb{R})$, $k = 1, 2$, where $P_k = P_k^2 = P_k^\dagger$ are orthogonal projections. Suppose further

$$(P_1 \otimes P_2)A = A,$$

i. e. for every x , $A(x, \circ)$ lies in the same subspace $\text{Ran}(P_2)$, and for every y , $A(\circ, y)$ lies in $\text{Ran}(P_1)$, where Ran denotes the range of a map. Now, we discretize V in the basis $e_i \otimes e_j$, where $e_i \in L^2(\mathbb{R})$, $i = 1, \dots, n$. To represent the potential exactly, we need $\text{Ran}(P_k) \subset \text{span}(\{e_1, \dots, e_n\})$. However, even if the potential is slowly varying (i. e. we can span $\text{Ran}(P_k)$ with few basis vectors, or grid points), the representation of the phase oscillations associated with large momenta requires the use of a fine grid (higher-dimensional basis). The interpretation of this assumption is that we *oversample* the potential, i. e. use a finer basis than necessary to represent it to sufficient accuracy. In that case, the projectors $(P_k)_{ij} = \langle e_i, P_k e_j \rangle$ are of low rank and we have

$$\begin{aligned} A_{ij} &= \langle e_i \otimes e_j, A \rangle \\ &= (P_1)_{ii'} (P_2)_{jj'} A_{i'j'}. \end{aligned}$$

Therefore, the matrix A is of $\text{rank}(A) \leq \min(\text{rank}(P_1), \text{rank}(P_2))$ and the SVD provides a data-sparse representation.

Notice that when doing Schrödinger quantum mechanics on $L^2(\mathbb{R}^2)$ this cannot be used as one needs to apply $\psi'_{ij} = A_{ij} \psi_{ij}$. However, when working with Hartree type approximations (see Section 1.4.3)

$$\psi = \psi^{(1)} \otimes \psi^{(2)},$$

one needs to evaluate partial traces

$$A_i^{(1)} = \sum_j \psi_j^{(2)*} A_{ij} \psi_j^{(2)}.$$

Calculating $A^{(1)}$ reduces from apply count n^2 to $2kn$ if one has a rank- k approximation of A .

1.2.2. \mathcal{H} Matrices

It is important to note that the rank of a matrix is basis-independent. The statement “a matrix is of low rank” is a very strong one; in particular, one “simple” matrix, the identity, has full rank. Transferring our starting point, that not all parts of interaction space interact with all others in a complex manner, to matrices, we reason that different subspaces interact with others in more or less “complicated” ways. This translates into different submatrices being of different rank. In other words, we split

the matrix into submatrices according to a certain *blocking* and perform a low rank approximation on each submatrix. This way, if large parts of the matrix are “simple,” we can obtain a data-sparse representation of the matrix.

Matrix Blockings Let us make precise, what we mean by blocking. Define $I_{n,m} = \{n, n+1, \dots, m\}$ and $I_n = I_{1,n}$. A blocking for the index set I_n of a $n \times n$ matrix is given by sets $L_i = I_{n_i, m_i}$, $R_i = I_{n'_i, m'_i}$, such that

$$(L_i \times R_i) \cap (L_j \times R_j) = \emptyset, \text{ if } i \neq j, \quad (1.4)$$

i. e. the blocks are disjoint, and

$$\bigcup_i (L_i \times R_i) = I_n \times I_n, \quad (1.5)$$

i. e. the blocking covers the whole matrix. Such a blocking $\{L_1 \times R_1, \dots\}$ splits the matrix into submatrices

$$A_i = A_{L_i, R_i} = (A_{ab})_{a=n_i, \dots, m_i, b=n'_i, \dots, m'_i}.$$

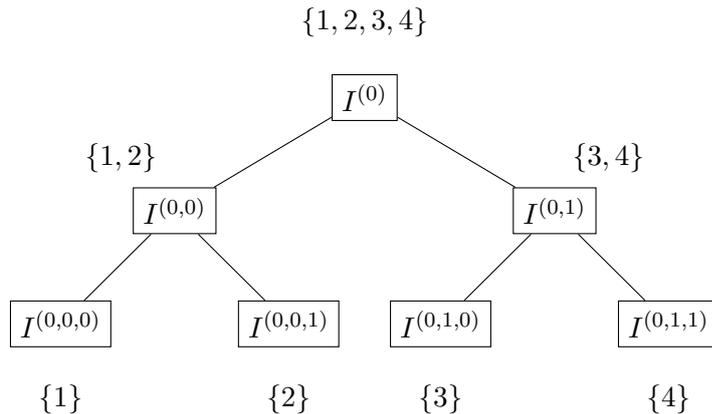
\mathcal{H} Blockings The notion of \mathcal{H} matrices [11] describes a class of blockings based on the idea that we can order the index set of the matrix in a physically meaningful way. The canonical example for an index set that can be ordered linearly is a one-dimensional grid on $[a, b]$, where the index $i = 1, \dots, n$ corresponds to a basis function located around $a + (b-a)(i-1/2)/n$. To introduce such so called \mathcal{H} blockings, we begin with the *index tree*. This is a representation of I_n in terms of recursive clustering. Every node of the tree is an index set $I_{a,b}$, the root of an index tree is the set I_n ; the leaves are the sets $I_{1,1}, \dots, I_{n,n}$. Every node satisfies the condition that it is either a leaf, or it is the union of all its children. We will denote a node in the index tree as

$$I^a = I^{(a_0, \dots, a_b)},$$

where $a_0 = 0$ and a_1, \dots, a_b describes the path, starting from the root, to travel to that node. We find it more convenient to index the first child with zero instead of one. For example, $I^{(0,0,2)}$ is the third child of the first child of the root; $I^{(0)} = I_n$. Hackbusch [11] introduces one particular index tree for $n = 2^m$, which will be used throughout this section: We split $I^{(0)} = I_n$ into $I^{(0,0)} = I_{1, n/2}$ and $I^{(0,1)} = I_{n/2+1, n}$, i. e. in half, and continue to do so until we reach the leaf level. That means, at the lowest level we have

$$I^{(0, a_1, \dots, a_m)} = \left\{ 1 + \sum_{k=1}^m a_k 2^{m-k} \right\},$$

which is the set containing the binary number $a_1 a_2 \dots a_m$. This index tree is exemplified for $n = 4$ in Fig. 1.1.

Figure 1.1.: Index tree for $n = 4$

The intuition behind this is the case of a local interaction, where far away regions interact “in a less complicated manner” than closely separated ones do. The term region translates into node in the index tree. The higher up a node is in the tree, the bigger the region it represents. Without any truncation, the interaction is represented on the leaf level (a full matrix). However, it might be sufficient to represent the interaction between far separated regions (off-diagonal elements) on a higher, that is, more coarse-grained, level further up in the index tree. In contrast, the interactions between nearby regions (close-to-diagonal elements) probably needs to be represented on a more fine-grained, lower level. Let us put this intuition into a definition.

Given an index tree I^a a \mathcal{H} blocking for I_n and this index tree is a blocking $\{L_1 \times R_1, \dots\}$ for I_n that satisfies $L_i = I^a$ for some a and $R_i = I^{a'}$ for some a' , where both a and a' are on the same level in the index tree. Hackbusch [11] describes the \mathcal{H} blocking P_2 for the index tree defined earlier. Starting at the root of the index tree on both sides of the matrix, the next level consists of four submatrices $I^{(0,0)} \times I^{(0,0)}$, $I^{(0,0)} \times I^{(0,1)}$, $I^{(0,1)} \times I^{(0,0)}$, $I^{(0,1)} \times I^{(0,1)}$. We obtain the P_2 blocking by placing the two off-diagonal blocks in it and “refining” the diagonal blocks. That means, we repeat this step for both $I^{(0,a_1)} \times I^{(0,a_1)}$ as roots. More precisely, following the two off-diagonal matrices at the first level, we place four off-diagonal matrices at the second level in the blocking: $I^{(0,a_1,0)} \times I^{(0,a_1,1)}$ and $I^{(0,a_1,1)} \times I^{(0,a_1,0)}$ for both $a_1 = 0, 1$. This way, we recursively refine the matrix near the diagonal until we arrive at the leaves and stop. This blocking is visualized in Fig. 1.2 for the case $n = 4$.

With this, we can define an $\mathcal{H}k$ matrix for a given \mathcal{H} blocking as a matrix A , where all submatrices defined by the blocking, A_i , are of maximal rank k . Analogously to the previous chapter, determining whether a matrix is $\mathcal{H}k$ for a given blocking is a straightforward task, accomplished by decomposing the submatrices; also optimal approximations (in the sense of Frobenius norm truncation error) exist by the Eckart-Young theorem. Let us quickly elaborate on this; suppose the submatrices A_i are approximated by rank- k matrices $A_{i,k}$, defined by the Eckart-Young theorem, giving

$I^{(0,0,0)} \times I^{(0,0,0)}$	$I^{(0,0,0)} \times I^{(0,0,1)}$	$I^{(0,0)} \times I^{(0,1)}$	
$I^{(0,0,1)} \times I^{(0,0,0)}$	$I^{(0,0,1)} \times I^{(0,0,1)}$		
$I^{(0,1)} \times I^{(0,0)}$		$I^{(0,1,0)} \times I^{(0,1,0)}$	$I^{(0,1,0)} \times I^{(0,1,1)}$
		$I^{(0,1,1)} \times I^{(0,1,0)}$	$I^{(0,1,1)} \times I^{(0,1,1)}$

Figure 1.2.: P_2 blocking exemplified for $n = 4$

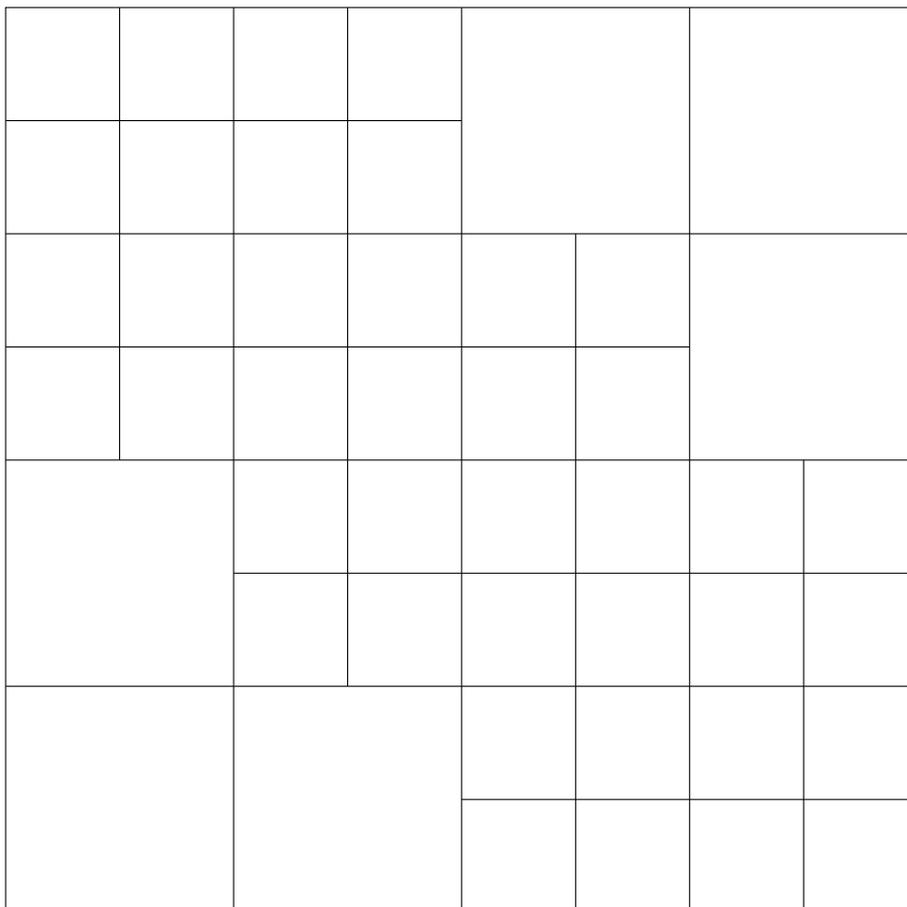


Figure 1.3.: P'_2 blocking exemplified for $n = 8$. Each square represents one submatrix in the blocking approximated by a rank- k matrix.

us the $\mathcal{H}k$ approximation A_k . Then

$$\|A - A_k\|_F^2 = \sum_i \|A_i - A_{i,k}\|_F^2,$$

and the optimality of the approximation in every submatrix guarantees optimality of the \mathcal{H} matrix truncation. Notice that condition (1.4) is essential here.

This concept can be adapted and generalized in an obvious manner. As one example we give the blocking P'_2 for the same index tree in Fig. 1.3. Whereas in P_2 we can find large blocks that have $i = j \pm 1$ index pairs, i. e. interactions between immediate neighbours, in P'_2 all blocks (except for the ones not approximated, that is, 1×1) correspond to well-separated regions. We expect the blocking P'_2 to be of greater use in practice.

Application Counts To investigate the applications count, that is, the number of complex multiplications required for $\psi'_i = A_{ij}\psi_j$, we recall that the application of a rank- k $n \times n$ matrix can be achieved with $2kn$ multiplications. To obtain the apply count, we sum over all blocks with their respective sizes and k fixed. By doing so, we calculate an upper bound as, for example, a 2×2 matrix of rank at least one can always be applied using its full representation without requiring additional resources. Analogously, if $k = 3$, any matrix with $n \leq 6$ can be applied faster without a low-rank approximation.

We begin with the P_2 blocking; denoting the upper bound for the apply count for an $n \times n$ $\mathcal{H}k$ matrix by $\text{Appc}_{P_2}(n, k)$, we have

$$\text{Appc}_{P_2}(n, k) = 2\text{Appc}_{P_2}\left(\frac{n}{2}, k\right) + 2\left(2k\frac{n}{2}\right),$$

where the first term accounts for the diagonal blocks, and the second term is the apply count for the two rank- k approximations of the off-diagonal blocks. The recursion stops with $\text{Appc}_{P_2}(1, k) = 1$, and we obtain

$$\text{Appc}_{P_2}(n, k) = 2kn \log_2(n) + n.$$

If we are able to refine the calculation by increasing n without having to also increase k this yields a significant improvement over n^2 . So, in calculating explicit approximations, we will always want to find a bound for the truncation error that can be controlled solely by k .

In the case P'_2 , at every level, the matrix is split into four matrices, two $\mathcal{H}k$ matrices and the two off-diagonal matrices. The applications count for such an $n \times n$ off-diagonal matrix can be obtained from

$$A_n = A_{n/2} + 3 \times 2k\frac{n}{2}.$$

With $A_1 = 1$ we have

$$A_n = 1 + 6k \sum_{a=0}^{\log_2 n - 1} 2^a = 1 + 6k(n - 1).$$

One easily verifies that

$$\begin{aligned} \text{Appc}_{P'_2}(n, k) &= 2\text{Appc}_{P'_2}\left(\frac{n}{2}, k\right) + 2\left(1 + 6k\left(\frac{n}{2} - 1\right)\right) \\ &= 6kn \log_2(n) + n + 2(1 - 6k)(n - 1), \end{aligned}$$

reproducing $O(kn \log n)$ scaling.

\mathcal{H} Matrix Approximations It is instructive to explicitly carry out a truncation to \mathcal{H} matrix format. In this section we closely follow [11]; however, we present the calculations in greater detail. Let us pick a local function $F(x, y) = f(x - y)$, where f is analytical on $[0, 1] \times [0, 1]$, discretized with respect to the basis functions $e_i = \sqrt{n}\chi_{[(i-1)/n, i/n]}$,⁴ $i = 1, \dots, n$. The prefactor is chosen such that the basis functions are normalized. We have

$$F_{ij} = n \int_{(i-1)/n}^{i/n} dx \int_{(j-1)/n}^{j/n} dy f(x - y).$$

We approximate this matrix by a \mathcal{Hk} matrix, where we want to compare different blockings. In this local and ordered basis, we verify that every block $L_i \times R_i$ corresponds to a rectangular subset of $B_i \subset [0, 1] \times [0, 1]$. We drop the index i and carry out the calculation remembering that we are restricting ourselves to one particular block $B = [x^l, x^r] \times [y^l, y^r]$.

The truncation of F to \mathcal{Hk} corresponds to the truncated Schmidt decomposition (that is, the truncated SVD)⁵ of $f(x, y) = f(x - y) = \sum_{\alpha=1}^k f_{\alpha}^{(1)}(x)f_{\alpha}^{(2)}(y)$ on B . We can obtain upper bounds on truncation errors by finding any such decomposition, without requiring it to be the (optimal) Schmidt decomposition. Following Hackbusch [11], we expand $f(x - y)$ around $y^c = (y^l + y^r)/2$, the center point of the y interval,

$$f(x - y) = \sum_{\alpha=0}^{\infty} \frac{(-)^{\alpha}}{\alpha!} f^{(\alpha)}(x - y^c)(y - y^c)^{\alpha}.$$

$f^{(\alpha)}$ denotes the α 'th derivative of f . This is such a decomposition into rank-one terms; therefore, we can bound the truncation error of F_{ij} by truncating this sum [11]

$$\begin{aligned} \epsilon_{ij}^k &\leq n \int_{(i-1)/n}^{i/n} dx \int_{(j-1)/n}^{j/n} dy \left| \sum_{\alpha=k}^{\infty} \frac{(-)^{\alpha}}{\alpha!} f^{(\alpha)}(x - y^c)(y - y^c)^{\alpha} \right| \\ &\leq \frac{1}{n} \max_{(x,y) \in B} \left| \sum_{\alpha=k}^{\infty} \frac{(-)^{\alpha}}{\alpha!} f^{(\alpha)}(x - y^c)(y - y^c)^{\alpha} \right| \\ &= \frac{1}{n} \max_{(x,y) \in B^c} \left| \sum_{\alpha=k}^{\infty} \frac{(-)^{\alpha}}{\alpha!} f^{(\alpha)}(x)y^{\alpha} \right| \\ &= \frac{1}{n} E_B^k, \end{aligned} \tag{1.6}$$

defining E_B^k . The region B^c is defined as

⁴The characteristic function χ_B of a set B evaluates to one if its argument is in B and zero else.

⁵Schmidt decomposition and SVD are two names for the same thing; in the context of tensor products one usually speaks of Schmidt decomposition, whereas SVD prevails when reasoning about matrices. The equivalence is given by $\mathbb{C}^{n \times n} \cong \mathbb{C}^n \otimes \mathbb{C}^n$.

$$\begin{aligned} (x, y) \in B &\Leftrightarrow (x - y^c, y - y^c) \in B^c \\ &\Rightarrow B^c = \left[x^l - y^c, x^r - y^c \right] \times \left[-(y^r - y^l)/2, (y^r - y^l)/2 \right]. \end{aligned}$$

With that, we can obtain the total truncation error in Frobenius norm by denoting $|L_i|$ to mean the number of elements in the set L_i

$$\begin{aligned} \left(\epsilon_F^k \right)^2 &\leq \sum_i \frac{|L_i| |R_i|}{n^2} \left(E_{B_i}^k \right)^2 \\ &\leq \max_i \left(E_{B_i}^k \right)^2. \end{aligned}$$

The second inequality is a well-known relation between Frobenius and maximum norms.

Let us investigate some examples. We begin with $f(x) = \log|x|$ together with P_2' blocking, the example presented in [11]. For positive x the derivatives read

$$f^{(\alpha)}(x) = (\alpha - 1)! (-)^{\alpha+1} x^{-\alpha}.$$

We have

$$E_B^k = \max_{(x,y) \in B^c} \left| \sum_{\alpha=k}^{\infty} \frac{1}{\alpha} \left(\frac{y}{x} \right)^\alpha \right|.$$

Let us start at the top level. There we have two $\mathcal{H}k$ matrices of half size, the diagonal ones, and two off-diagonal blocks. We only need to care about subblocks of the off-diagonal blocks at the first level since the subblocks of the diagonal ones correspond to blocks with boundaries x^l, x^r, y^l, y^r half as big as on the first level (plus the same blocks shifted by one-half). This means, the maximum will always go over the same values of y/x for $(x, y) \in B^c$.

The off-diagonal blocks contain four subblocks, three are represented as rank- k matrices, one (the one touching the diagonal) continues to be refined. Following a similar reasoning, we can restrict our attention to the six off-diagonal subblocks of the off-diagonal subblocks at the first level. By the symmetry of F , we only need to look at three of these. To make this more clear, we give them explicitly

$$\begin{aligned} B_1 &= \left[\frac{1}{2}, \frac{3}{4} \right] \times \left[0, \frac{1}{4} \right], B_1^c = \left[\frac{3}{8}, \frac{5}{8} \right] \times \left[-\frac{1}{8}, \frac{1}{8} \right], \\ B_2 &= \left[\frac{3}{4}, 1 \right] \times \left[0, \frac{1}{4} \right], B_2^c = \left[\frac{5}{8}, \frac{7}{8} \right] \times \left[-\frac{1}{8}, \frac{1}{8} \right], \\ B_3 &= \left[\frac{3}{4}, 1 \right] \times \left[\frac{1}{4}, \frac{1}{2} \right], B_3^c = \left[\frac{3}{8}, \frac{5}{8} \right] \times \left[-\frac{1}{8}, \frac{1}{8} \right]. \end{aligned}$$

We quickly verify that on the domain of the max function B_i^c the blocks one and three correspond to $y/x \in [-1/3, 1/3]$, whereas block two corresponds to $y/x \in [-1/5, 1/5]$ and obtain

$$\epsilon_F^k \leq \max_{x \in [-\frac{1}{3}, \frac{1}{3}]} \left| \sum_{\alpha=k}^{\infty} \frac{1}{\alpha} x^\alpha \right| = \sum_{\alpha=k}^{\infty} \frac{1}{\alpha} \left(\frac{1}{3}\right)^\alpha.$$

Firstly, we note that this result does not depend on n ; this means, we can adjust the truncation error by picking k and then refine the discretization (for example, since the dynamics simulated requires it), *without having to simultaneously increase the rank*. Therefore, we have $O(n \log n)$ scaling in this example. Furthermore, we notice the scaling with respect to k as $3^{-k}/k$, similarly to the result derived in [11]. This convergence is very fast, meaning the method we picked to represent F matches the structure of F well.

Can we extend this result to P_2 blocking? Not in a straight-forward manner. One quickly realizes that on the off-diagonal blocks the max is to be taken over $y/x \in [-1, 1]$ and the series diverges. This is the aforementioned issue inherent to P_2 blocking. Big submatrices (even at the topmost level) contain elements that describe next-neighbour interactions. We cannot employ (1.6) as this requires the function to be L^∞ .

Instead of trying to find another bound, we investigate this numerically. For this, we set up the matrix using an analytical result for the integral. Afterwards the matrix is truncated to $\mathcal{H}k$ for both P_2 and P'_2 blocking; we plot the truncation error in Frobenius norm against the rank k . Our results are presented in Fig. 1.4. As expected, the blocking P'_2 performs well, giving us an error estimate solely based on k , independent of n . Furthermore, the errors using P'_2 blocking are lower than using P_2 . Finally and most importantly, the P_2 errors cannot be bounded by k . Refining the discretization means the rank has to be increased as well; we do *not* get $O(n \log n)$ scaling.

Arithmetic As we have seen, matrix-vector products, as well as storage, can be achieved with *almost linear* scaling. However, the original paper [11] points out that a wider class of operations scales preferably. Addition of two $\mathcal{H}1$ matrices, followed by truncation to $\mathcal{H}1$ (note that matrices of bounded rank do not form a linear space) scales like $O(n \log n)$; multiplication (and following truncation) of two $\mathcal{H}1$ matrices is in $O(n \log^2 n)$. These operations inherit linear scaling from their corresponding counterparts in low-rank matrices with an additional dependence on $\log n$, paying for the refined representation around the matrix diagonal.

Interestingly, this even extends to inverses, in general a problem in $O(n^3)$. $\mathcal{H}1$ inversion with following truncation can be achieved with $O(n \log^2 n)$ operations.

Outlook In [12] \mathcal{H} matrices are applied to high-dimensional problems, where the matrices being approximated correspond to unfoldings of higher-order tensors. Hack-

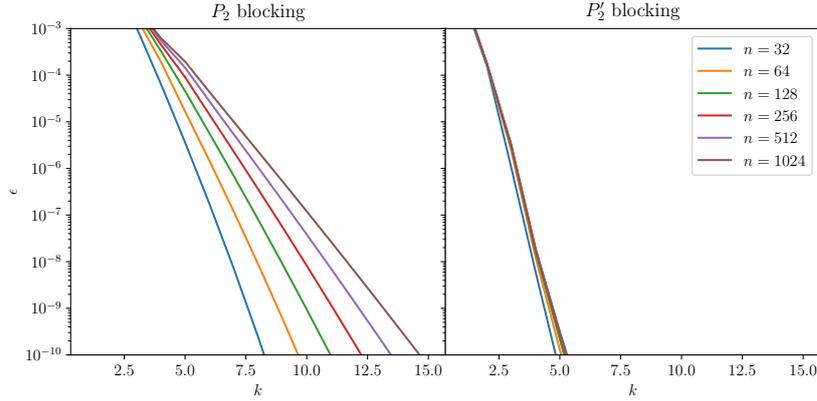


Figure 1.4.: Frobenius norm error of the truncation to $\mathcal{H}k$ format of the kernel $\log|x - y|$ discretized in a normed, piecewise constant basis, for two different blockings

busch and Khoromskij [13] show that for problem size n^d in d dimensions matrix-vector operations scale like $O(n^d \log^{d+1} n^d)$.

Gavrilyuk, Hackbusch and Khoromskij [14] use the technique to efficiently calculate the matrix exponential, which can be represented by a sum of resolvents. These in turn contain \mathcal{H} matrix structure for a certain class of elliptic operators.

1.2.3. Product Representations

It is an interesting question to be asked, whether one can find an optimal blocking (where what is meant by “optimal” remains open for now), given a certain matrix A . As the \mathcal{H} blocking scheme is a special one, when dealing with such an optimization problem, it seems reasonable to enlarge the domain in order to arrive at a formulation that we can work with.

We can cast \mathcal{H} matrices and low-rank representations into the same scheme, viewing them as *product representations* of matrices

$$A = UV^\dagger, \quad A \in \mathbb{C}^{m \times n}, \quad U \in \mathbb{C}^{m \times k}, \quad V \in \mathbb{C}^{n \times k},$$

where the representation is in fact data-sparse if the number of non-zeros in U and V is lower than the number of non-zeros in A . That is, we can use this product representation to translate between the terms data-sparse and sparse. For an SVD, the matrices U and V are full and the representation is data-sparse if

$$k(m + n) < mn,$$

or, for $m = n$, if $k < n/2$. This is consistent with applications count $2kn$, which beats the full applications count n^2 for $k < n/2$.

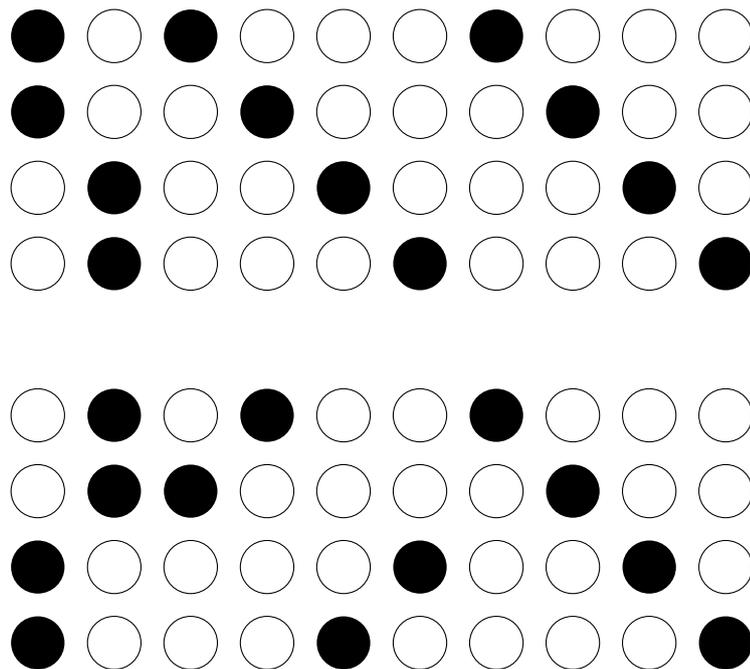


Figure 1.5.: Matrices U (top) and V (bottom) for a 4×4 P_2 $\mathcal{H}1$ matrix. A filled circle represents a non-zero entry.

For \mathcal{H} matrices, U and V contain structure; they are “masked.” By masked, we mean sparse in an a priori defined way; for example a banded matrix is masked. To illustrate that for \mathcal{H} matrices, let us go back to the blocking $P_2(4)$ as presented in Fig. 1.2 and set the rank of all submatrices to one. Then U and V satisfy, up to exchanging columns simultaneously, the conditions given in Fig. 1.5. The first two columns represent the off-diagonal 2×2 blocks as rank-1 matrices. The next four columns give the off-diagonal elements in the diagonal 2×2 blocks, and the last four columns represent the diagonal entries.

Let us cast this into a definition. A product scheme for a matrix in $\mathbb{C}^{m \times n}$ of size k is given by two mask matrices $M^U \in \{0, 1\}^{m \times k}$ and $M^V \in \{0, 1\}^{n \times k}$. A product representation of the matrix $A \in \mathbb{C}^{m \times n}$ consists of $U \in \mathbb{C}^{m \times k}$ and $V \in \mathbb{C}^{n \times k}$, such that

$$U \circ M^U = U, \quad V \circ M^V = V, \quad A = UV^\dagger, \quad (1.7)$$

where \circ denotes element-wise multiplication. This is slightly more general than a \mathcal{H} matrix in that it drops the conditions of disjointness (1.4) and full coverage (1.5). The latter only implies that we allow product schemes for matrices, where we know a priori certain entries are zero. The former’s implications, however, are not clear. Also, product representations are more general in that they do not require each submatrix to be of equal rank and submatrices need not be $I_{n,m} \times I_{n',m'}$ blocks. This poses the question, whether we can gain anything from allowing these more general schemes.

Can we find an optimal approximation given a scheme (M^U, M^V) and a matrix A ? If so, does this approximation satisfy any uniqueness conditions? Can we even find an apply count-optimal scheme (M^U, M^V) given a matrix A and a truncation error?

The former is an optimization problem given by the loss function

$$L : \mathbb{C}^{m \times k} \times \mathbb{C}^{n \times k} \rightarrow \mathbb{R},$$

$$(U, V) \mapsto \left\| A - (U \circ M^U) (V \circ M^V)^\dagger \right\|_F^2.$$

In the disjoint case, this splits up into multiple SVDs as before in the \mathcal{H} matrix case. However, if the submatrices are not disjoint, this represents multiple “coupled SVDs.”

Let us investigate this situation with a simple example for $m = n = 2$, given by the mask matrices

$$M^U = M^V = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}.$$

Denote the entries of A , U and V by

$$A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}, \quad U = \begin{pmatrix} u_1 & x \\ u_2 & 0 \end{pmatrix}, \quad V = \begin{pmatrix} v_1 & 1 \\ v_2 & 0 \end{pmatrix},$$

where we have $U \circ M^U = U$ as well as $V \circ M^V = V$. Let us see if we can represent A exactly; this gives us the set of equations

$$\begin{aligned} a_{11} &= u_1 v_1^* + x, \\ a_{12} &= u_1 v_2^*, \\ a_{21} &= u_2 v_1^*, \\ a_{22} &= u_2 v_2^*. \end{aligned}$$

By choosing x , we can always satisfy the first equation. This means, the problem we are looking at is an SVD, where we do not care about the error made in one particular entry. We need to distinguish two cases now; firstly $a_{22} \neq 0$. Then, choosing

$$\begin{aligned} u_1 &= a_{12}, \\ u_2 &= a_{22}, \\ v_1 &= a_{21}^* / a_{22}^*, \\ v_2 &= 1, \end{aligned}$$

we have $A = UV^\dagger$.

Secondly, let us look at $a_{22} = 0$. We now have the set of equations

$$\begin{aligned} a_{12} &= u_1 v_2^*, \\ a_{21} &= u_2 v_1^*, \end{aligned}$$

with the constraint $u_2 = 0 \vee v_2 = 0$, which can only be satisfied if $a_{12} = 0 \vee a_{21} = 0$.

In summary, all matrices that satisfy $a_{22} \neq 0 \vee a_{12} = 0 \vee a_{21} = 0$ can be represented exactly, i.e. all *except for a set of measure zero*. However, for any matrix with $a_{22} = 0 \wedge a_{12} \neq 0 \wedge a_{21} \neq 0$ and every $\epsilon > 0$, we can find an approximation

$$\begin{aligned} u_1 &= a_{12}, \\ u_2 &= \epsilon, \\ v_1 &= a_{21}^*/\epsilon, \\ v_2 &= 1 \end{aligned}$$

that satisfies $\|A - UV^\dagger\| \leq \epsilon$. This result is wildly different from Eckart-Young's theorem in that we cannot find an optimal approximation, but we can find an arbitrarily good approximation. However, in improving the accuracy of the approximation, the entries in the factor matrices diverge.

This simple example shows that dropping the disjointness (1.4) leads to an ill-posed approximation problem. This is not to say that such approximations cannot be useful, however it appears unlikely that further following along this path yields numerically useful results.

1.3. Higher Order Tensors

After studying matrix product representations, most notably the singular value decomposition, we will now go on to higher dimensions and study analogues.

1.3.1. Notation

In the context of this text a *tensor* is a vector, an element of a linear space. The linear spaces we are concerned with are \mathbb{C}^N with the inner product

$$\langle v, w \rangle = \sum_{i=1}^N v_i^* w_i.$$

A higher-order tensor is to be understood as a vector in a tensor product space

$$\mathbb{C}^N \cong \mathbb{C}^{n_1} \otimes \mathbb{C}^{n_2} \otimes \dots \otimes \mathbb{C}^{n_d}.$$

We call the tuple (n_1, \dots, n_d) the *shape* of the tensor, which is often denoted as $n_1 \times \dots \times n_d$. The number d is called the *order* of the tensor. More precisely, a tensor is a vector in the linear space \mathbb{C}^N together with the tuple (n_1, \dots, n_d) , $N = \prod_{i=1}^d n_i$.

A scalar is a tensor of shape (1) , a vector has shape (n) and a matrix is a tensor with shape (n, m) .

Obviously, the space \mathbb{C}^N allows us to identify tensors of different shapes. For example a matrix of shape (n, m) can be seen as a vector of shape (nm) ; this process is called *unfolding*, or, in the special case of identifying a higher-order tensor $d > 2$ with a matrix $d = 2$, *matricization*. As this identification is not unique, using these terms will always require us to make precise, how the unfolding works.

The inner product introduced above induces a norm on every tensor space; for matrices this norm coincides with the Frobenius norm. Therefore, we will denote this norm as $\|\circ\|_F$.

1.3.2. Generalizing Rank

Let us quickly recall the SVD of a matrix A

$$A_{i_1 i_2} = \sum_{\alpha=1}^k \sigma_{\alpha} a_{i_1 \alpha}^{(1)} a_{i_2 \alpha}^{(2)},$$

satisfying

$$\sum_i a_{i\alpha}^{(f)*} a_{i\beta}^{(f)} = \delta_{\alpha\beta}, \quad (1.8)$$

for $f = 1, 2$, i. e. the factor matrices are unitary. This follows from the fact that AA^\dagger and $A^\dagger A$ are self-adjoint matrices. The rank of the matrix is the smallest number k , for which we can find such a representation. Recall that this decomposition is unique up to a unitary transformation on every degenerate subspace and reordering the columns of $a^{(f)}$ together with the singular values σ . This uniqueness only follows due to the fact that we impose unitarity of the factor matrices. If we were to only require that each column $(a_{i\alpha}^{(f)})_i$ be normalized to one, but omit the orthogonality of the columns, we can, denoting $U = a^{(1)}$, $V = a^{(2)*}$, choose

$$U' = U\sigma T\sigma'^{-1}, \quad V' = VT,$$

for a unitary transformation T , where we choose the diagonal matrix σ' such that the columns of U' are normalized and arrive at another decomposition

$$U'\sigma'V'^{\dagger} = U\sigma T T^{\dagger} V^{\dagger} = U\sigma V^{\dagger}.$$

The straight-forward generalization to higher-order tensors is given by the *canonical decomposition* (CP) [15–17]

$$A_{i_1 \dots i_d} = \sum_{\alpha=1}^k \sigma_{\alpha} a_{i_1 \alpha}^{(1)} \dots a_{i_d \alpha}^{(d)}, \quad (1.9)$$

where we require

$$\sum_i a_{i\alpha}^{(f)*} a_{i\alpha}^{(f)} = 1 \quad (1.10)$$

for all α, f . Notice that we do not require the factor matrices to be unitary. Analogously to the matrix case we define the *rank of the tensor* as the smallest number k , for which such a representation exists.

The CP has a wide array of interesting properties (see, for example, [18] and references therein), most notably *it is hard* to find a decomposition with $k = \text{rank}(A)$ (more precisely NP-hard). It is even NP-hard to find the rank of a general tensor.

Moreover, CPs for certain tensor shapes satisfy stronger uniqueness conditions, i. e. they are unique without imposing any constraints. Kruskal [19] proves (among others) the following theorem for order-3 tensors A . Suppose every $k^{(1)}$ ($k^{(2)}, k^{(3)}$) columns of the factor matrix $a^{(1)}$ ($a^{(2)}, a^{(3)}$) are linearly independent; suppose further $k^{(1)} + k^{(2)} + k^{(3)} \geq 2 \text{rank}(A) + 2$. Then the canonical decomposition of A in terms of $a^{(f)}$ is unique up to reordering and applying phases to the columns in the factor matrices.

These properties are related to the fact that we can no longer impose a unitarity requirement on the factor matrices. That this cannot be the case, can be seen immediately from a dimensionality argument. With real dimension of $U(n)$ equal to n^2 and real dimension of the space of $(\mathbb{C}^n)^{\otimes d}$ tensors equal to $2n^d$, we see that the space spanned by decompositions with unitary factors has real dimension bounded by $dn^2 + n$ and cannot possibly cover the full space for $d > 2$. For $d = 2$, the matrix case, the additional n dimension correspond to the freedom of choosing a phase ($U(1)$ transformation) on every column of the factor matrices. If we could impose unitarity, the rank of any $n \times \dots \times n$ tensor would be bounded by n (as is the case for an $n \times n$ matrix); however, in reality, even finding bounds for the ranks of general tensors is a hard endeavour.

As for an analogue of the approximation theorem by Eckart-Young; it does not exist [18]. For a given tensor A and a rank k , there does not necessarily exist a rank- k approximation A_k optimizing the Frobenius norm error. Kolda and Bader [18] introduce the notion of a *degenerate tensor* as a tensor that can be approximated arbitrarily well by a tensor of lower rank and give the following third order tensor of rank three as an example:

$$A_{i_1 i_2 i_3} = a_{i_1 1}^{(1)} a_{i_2 1}^{(2)} a_{i_3 2}^{(3)} + a_{i_1 1}^{(1)} a_{i_2 2}^{(2)} a_{i_3 1}^{(3)} + a_{i_1 2}^{(1)} a_{i_2 1}^{(2)} a_{i_3 1}^{(3)},$$

where the columns of the factor matrices are linearly independent. This tensor can be approximated arbitrarily well by the rank-two tensor

$$A_{i_1 i_2 i_3}^\epsilon = \frac{1}{\epsilon} \left(\left(a_{i_1 1}^{(1)} + \epsilon a_{i_1 2}^{(1)} \right) \left(a_{i_2 1}^{(2)} + \epsilon a_{i_2 2}^{(2)} \right) \left(a_{i_3 1}^{(3)} + \epsilon a_{i_3 2}^{(3)} \right) - a_{i_1 1}^{(1)} a_{i_2 1}^{(2)} a_{i_3 1}^{(3)} \right),$$

as we verify quickly from

$$A_{i_1 i_2 i_3} - A_{i_1 i_2 i_3}^\epsilon = \epsilon \left(a_{i_1 1}^{(1)} a_{i_2 2}^{(2)} a_{i_3 2}^{(3)} + a_{i_1 2}^{(1)} a_{i_2 1}^{(2)} a_{i_3 2}^{(3)} + a_{i_1 2}^{(1)} a_{i_2 2}^{(2)} a_{i_3 1}^{(3)} \right) + \epsilon^2 a_{i_1 2}^{(1)} a_{i_2 2}^{(2)} a_{i_3 2}^{(3)}.$$

Notice that the entries of the factor matrices of A^ϵ diverge as $\epsilon \rightarrow 0$, which reminds us of the situation we encountered in Section 1.2.3.

1.3.3. Higher Order SVD

As we have seen, a rank decomposition of a general tensor is a hard endeavour; low-rank approximations are plagued by ill-posedness of the optimization problem. This, besides the mathematical peculiarity, means that we cannot make use of it in a numerical context—at least not in a straight-forward setting.

However, there are other decompositions being used with great success [18]. To take a step in that direction, let us go back to the issue of imposing unitarity of the factor matrices. This possibility sets the SVD aside from the CP. So, for a moment, let us assume, we are dealing with an order- d tensor that has a decomposition (1.9) in terms of unitary factors

$$\sum_i a_{i\alpha}^{(f)*} a_{i\beta}^{(f)} = \delta_{\alpha\beta}.$$

Intuitively, we expect there to be an algorithm similar to the SVD to reproduce the factors and “singular values” from the tensor elements. To obtain the SVD of a matrix A , we diagonalize AA^\dagger . For higher order tensors, we find the analogue of matrix multiplication in tensor contraction

$$[A]_{ii'}^{(f)} = \sum_{i_1} \dots \widehat{\sum_{i_f}} \dots \sum_{i_d} A_{i_1 \dots i_f \dots i_d} A_{i_1 \dots i' \dots i_d}^*,$$

where the hat marks ellipsis of this sum. Notice that for $d = 2$, $[A]^{(1)} = AA^\dagger$ and $[A]^{(2)T} = A^\dagger A$. This summation corresponds to tracing out, or integrating out, all but one degree of freedom in $A \otimes A^*$.

Now, since we assume there to be a unitary decomposition of A we obtain

$$\begin{aligned} [A]_{ii'}^{(f)} &= \sum_{\alpha=1}^k \sigma_\alpha \sum_{\alpha'=1}^k \sigma_{\alpha'} \left(\sum_{i_1} a_{i_1 \alpha}^{(1)} a_{i_1 \alpha'}^{(1)*} \right) \dots \left(a_{i_\alpha}^{(f)} a_{i' \alpha'}^{(f)*} \right) \dots \left(\sum_{i_d} a_{i_d \alpha}^{(d)} a_{i_d \alpha'}^{(d)*} \right) \\ &= \sum_{\alpha=1}^k \sigma_\alpha^2 a_{i\alpha}^{(f)} a_{i'\alpha}^{(f)*} \end{aligned}$$

and have the singular values as well as the factor matrices readily available by diagonalizing the Hermitian matrices $[A]^{(f)}$.

This procedure is known as *higher order SVD* [20, 21] and can be applied to general tensors. Let us investigate what happens if A is not decomposable in terms of unitary matrices. $[A]^{(f)}$ is still a Hermitian matrix, and we can diagonalize it obtaining the matrix of eigenvectors $a_{i\alpha}^{(f)}$, where we exclude eigenvectors corresponding to eigenvalue zero (strictly speaking $a^{(f)}$ is semi-unitary). We can then use these matrices, somehow implying they “improve” the tensor, as basis transformations

$$\sigma_{\alpha_1 \dots \alpha_d} = \sum_{i_1} \dots \sum_{i_d} a_{i_1 \alpha_1}^{(1)*} \dots a_{i_d \alpha_d}^{(d)*} A_{i_1 \dots i_d}.$$

Tentatively, the transformed tensor is named σ as, in the case of a unitarily decomposable tensor we have the diagonal tensor

$$\sigma_{\alpha_1 \dots \alpha_d} = \delta_{\alpha_1 \alpha_2 \dots \alpha_d} \sigma_{\alpha_1}$$

consisting of the “singular values” of A . In the general case we call σ the *core tensor* and arrive at the *Tucker decomposition* [20, 22]

$$A_{i_1 \dots i_d} = \sum_{\alpha_1} \dots \sum_{\alpha_d} \sigma_{\alpha_1 \dots \alpha_d} a_{i_1 \alpha_1}^{(1)} \dots a_{i_d \alpha_d}^{(d)}. \quad (1.11)$$

Let us take a step back. We could have taken any d unitary matrices and transformed the tensor A to arrive at a representation (1.11). In fact, we will call any such representation Tucker decomposition.

Definitions To discuss properties of the HOSVD, we first introduce the notions rigorously. We define the *Tucker representation* as a mapping

$$\text{Tucker}(\sigma, a^{(f)})_{i_1 \dots i_d} = \sum_{j_1} \dots \sum_{j_d} a_{i_1 j_1}^{(1)} \dots a_{i_d j_d}^{(d)} \sigma_{j_1 \dots j_d},$$

where the boundaries of the sums are given by the shape of σ . $(\sigma, a^{(f)})$ is called *Tucker decomposition* of a tensor A if $A = \text{Tucker}(\sigma, a^{(f)})$. Obviously, setting $\sigma = A$ and $a^{(f)} = \text{Id}$ yields a trivial Tucker decomposition.

To make the analogy between SVD and HOSVD clearer, we define the f -rank of a tensor.⁶ Denoted as $\text{rank}_f(A)$, the f -rank (1-rank, \dots , d -rank) of A is given by

$$\text{rank}_f(A) = \text{rank}([A]^{(f)}).$$

Analogously to viewing HOSVD as a more straight-forward generalization of the SVD than the CP, the f -rank yields a straight-forward generalization of matrix column and row rank. SVD reveals the rank of a matrix. HOSVD reveals the f -rank of a

⁶Usually this is introduced as n -rank; however, to avoid confusion with the dimension of the single-particle space we use f -rank in this text.

tensor as the shape of σ is given by $\text{rank}_1(A) \times \cdots \times \text{rank}_d(A)$. A low-rank tensor is a tensor of shape $n_1 \times \cdots \times n_d$ that has $\text{rank}_f(A) \ll n_f$, and HOSVD provides a way to compress it. Notice that any Tucker decomposition $A = \text{Tucker}(\sigma, a^{(f)})$ is rank-revealing, in so far as we have $\text{rank}_f(A) \leq r_f$ for σ of shape $r_1 \times \cdots \times r_f$. We note that the rank of a tensor is bounded by $\text{rank}(A) \geq \text{rank}_f(A)$.

Furthermore, we note that the core tensor σ generated by the algorithm described above is *all-orthogonal*. By all-orthogonal we mean

$$[\sigma]_{ij}^{(f)} = 0, \text{ if } i \neq j.$$

Finally, we define an HOSVD a tensor A as a Tucker decomposition $A = \text{Tucker}(\sigma, a^{(f)})$, where σ is all-orthogonal and $a^{(f)}$ are semi-unitary. As de Lathauwer, de Moor, and Vandewalle [21] point out, this definition uniquely defines the HOSVD up to freedom of choice as in the SVD. It is instructive to go through the proof of this statement in detail.

Uniqueness of HOSVD Following [21] we claim that any two HOSVDs of a tensor A are related by reordering the ranks and a unitary transformation on degenerate subspaces, i. e. HOSVD is unique up to the freedom one has in choosing an orthonormal eigenbasis of $[A]^{(f)}$, or up to the freedom one has in choosing the SVD of a matrix. This means, we can indeed talk about *the* HOSVD.

We start by noting that for unitary $a^{(f)}$ the following relation holds:

$$\left[\text{Tucker}(\sigma, a^{(f)}) \right]^{(f)} = a^{(f)} [\sigma]^{(f)} a^{(f)\dagger}.$$

Assume now two different HOSVDs of A are given by

$$A = \text{Tucker}(\sigma, a^{(f)}) = \text{Tucker}(\sigma', b^{(f)}).$$

Hence

$$[A]^{(f)} = a^{(f)} [\sigma]^{(f)} a^{(f)\dagger} = b^{(f)} [\sigma']^{(f)} b^{(f)\dagger}$$

This, together with the fact that σ and σ' are all-orthogonal, implies that $[\sigma]^{(f)}$ and $[\sigma']^{(f)}$ are identical up to reordering and a unitary transformation on degenerate subspaces. So we can safely assume $[\sigma]^{(f)} = [\sigma']^{(f)}$.

Furthermore, we have

$$\sigma' = \text{Tucker}(\sigma, b^{(f)\dagger} a^{(f)}) = \text{Tucker}(\sigma, c^{(f)}),$$

introducing $c^{(f)}$. Plugging these two results together gives us the condition

$$\begin{aligned} \left[\text{Tucker}(\sigma, c^{(f)}) \right]^{(f)} &= [\sigma]^{(f)}, \\ c^{(f)} [\sigma]^{(f)} c^{(f)\dagger} &= [\sigma]^{(f)}. \end{aligned}$$

In other words, $c^{(f)}$ is nothing but a unitary transformation on each degenerate subspace ($U(1) \oplus \dots \oplus U(1)$ in case the diagonal entries of $[\sigma]^{(f)}$ are all different). Since $\sigma' = \text{Tucker}(\sigma, c^{(f)})$, uniqueness of the HOSVD follows.

SVD is a Special Case of HOSVD An HOSVD of a matrix A can be written as $A = U\sigma V^\dagger$, where U and V are unitary and σ is all-orthogonal. Notice that a matrix σ is all-orthogonal if $\sigma\sigma^\dagger$ as well as $\sigma^\dagger\sigma$ are diagonal. This implies, using the SVD on σ , where we choose the matrix of left-singular values to be the identity, that

$$\sigma = dS^\dagger,$$

with a diagonal d and unitary S satisfying $d^\dagger d = \sigma\sigma^\dagger$ and $\sigma^\dagger\sigma S = Sd^\dagger d$. With $\sigma^\dagger\sigma = P^T\sigma\sigma^\dagger P$, where P is a permutation, we have

$$S = P^T T, \text{ where } [d^\dagger d, T] = 0.$$

That means, σ is the diagonal matrix of singular values up to a permutation of the columns and a unitary transformation on degenerate subspaces. In summary, for a HOSVD of a matrix we can impose the additional condition, namely that σ is diagonal with positive entries, by choosing

$$V' = VP^T T \Rightarrow A = UdV'^\dagger,$$

and arrive at the SVD.

Notice that we only have this additional freedom for $d = 2$ as a superdiagonal σ for higher dimensions implies that the HOSVD is a rank decomposition and hence $\text{rank}(A) \leq n$ if A is of shape $n \times \dots \times n$, which is not true in general.

POTFIT A well-known use of HOSVD is referred to as *POTFIT* [23] and yields gains exactly when the tensor A does not have full f -rank. In fact, we already discussed POTFIT for the case $d = 2$ in Section 1.2.1. We assume A to represent an *oversampled* potential, i. e. a slowly varying potential discretized in a too fine grid. This situation arises in practice when dealing with potential energy surfaces in chemical physics. To capture the dynamics happening on that potential a fine grid is necessary; to capture the structure of the potential a much less fine-grained representation would be sufficient. Analogously to Section 1.2.1 we can define projectors P_f , $f = 1, \dots, d$ on the d degrees of freedom and see that $\text{rank}[A]^{(f)} = \text{rank}(P_f) = r_f$. This implies, the tensor has f -rank r_f and can be expressed as

$$A_{i_1 \dots i_d} = \sum_{\alpha_1=1}^{r_1} \dots \sum_{\alpha_d=1}^{r_d} \sigma_{\alpha_1 \dots \alpha_d} a_{i_1 \alpha_1}^{(1)} \dots a_{i_d \alpha_d}^{(d)}.$$

Again, we cannot make use of this data-sparse representation if we are working with unconstrained correlations. If, however, we limit correlations, e. g.

$$\psi = \psi^{(1)} \otimes \dots \otimes \psi^{(f)},$$

the calculation of

$$A_{i_1}^{(1)} = \sum_{i_2 \dots i_d} A_{i_1 \dots i_d} \psi_{i_2}^{(2)*} \psi_{i_2}^{(2)} \dots \psi_{i_d}^{(d)*} \psi_{i_d}^{(d)}$$

can be sped up by transforming $\psi^{(f)}$ by $a_{i_\alpha}^{(f)}$. We obtain an applications count scaling like $\prod_i r_i$ instead of $\prod_i n_i$.

However, applying the HOSVD to a high-dimensional tensor can be computationally very demanding as the calculation of the matrices $[A]^{(f)}$ has applications count $n_f^2 \prod_{i \neq f} n_i$; this problem can be overcome using *multigrid POTFIT* [24], where the potential is represented on both a fine and a course grid; MGPF introduces an additional truncation error, which however appears to be negligible in practical cases.

Approximation Problem in HOSVD Analogously to the SVD and the CP, we ask the question: Given an $n_1 \times \dots \times n_d$ tensor A and f -ranks r_f , can we find an optimal approximation A_{opt} satisfying $\text{rank}_f(A_{\text{opt}}) = r_f$? We define optimality as in the Eckart-Young theorem as

$$\|A - A_{\text{opt}}\|_F = \min_{\text{rank}_f(\tilde{A})=r_f} \|A - \tilde{A}\|_F.$$

Kroonenberg [25] shows that a unique optimal solution exists in the case $d = 3$. The proof starts by showing that an optimal core tensor can be given for any unitary factor matrices $a^{(i)}$, which implies, the optimization is to be taken over the factor matrices only. These are unitary, therefore form a compact space and an optimal solution exists. This proof generalizes to higher d [26]; however, it remains unclear in which cases a unique optimal approximation exists.

We note that in the matrix case we can order the singular values by magnitude and simply truncate the expansion if σ_i falls below a threshold. The higher order singular values form a tensor and cannot be ordered by magnitude. However, the eigenvalues $\lambda_\alpha^{(f)}$ of $[A]^{(f)}$ corresponding to the α 'th column of $a^{(f)}$ give upper bounds on the magnitude of the core tensor elements [21]

$$|\sigma_{\alpha_1 \dots \alpha_d}|^2 \leq \min\{\lambda_{\alpha_1}^{(1)}, \dots, \lambda_{\alpha_d}^{(d)}\}.$$

Ordering the eigenvalues $\lambda_\alpha^{(f)}$ by magnitude, we can truncate the Tucker representation at $\alpha_f = 1, \dots, r_f$ denoting the resulting tensor as $A_{r_1 \dots r_d}$. We obtain [21]

$$\|A - A_{r_1 \dots r_d}\|_F^2 \leq \sum_{\alpha_1=r_1+1}^{n_1} \lambda_{\alpha_1}^{(1)} + \dots + \sum_{\alpha_d=r_d+1}^{n_d} \lambda_{\alpha_d}^{(d)}, \quad (1.12)$$

an equation analog to the truncated SVD (1.3), only we cannot give an equality. Finding the optimal approximation is, in general, achieved by optimization techniques,

starting from a truncated HOSVD. This happens using *alternating least squares* (ALS) methods, which fix all but one of the factor matrices. The best-known algorithm for this problem is referred to as *higher-order orthogonal iteration* (HOOI) [26]. It is interesting to note that the cost function

$$f(\tilde{A}) = \left\| A - \tilde{A} \right\|^2$$

can have multiple local minima on the space of rank- (r_1, \dots, r_d) tensors in the higher order case; that is, optimization techniques are not guaranteed to converge to a global optimum [26].

Optimization, however, may not be necessary as we have the bound [24, 27]

$$\frac{1}{d-1} \Lambda \leq \|A - A_{\text{opt}}\|_F^2 \leq \|A - A_{r_1 \dots r_d}\|_F^2 \leq \Lambda,$$

where Λ is the bound from (1.12). This means that the truncated HOSVD is already fairly close to the optimal approximation. Notice that for $d = 2$ we reproduce the Eckart-Young theorem.

Notes on Scaling To store a tensor $\in (\mathbb{C}^n)^{\otimes d}$ in full, one requires n^d complex numbers. Storing this tensor in the Tucker format with prescribed f -rank r for all f requires storage of d factor matrices of size $n \times r$ and one core tensor $\in (\mathbb{C}^r)^{\otimes d}$. In total, storage scales like $O(dnr + r^d)$. The implication is that, as opposed to the CP scheme, where a rank- r tensor requires storage $O(dnr)$, the Tucker decomposition still exhibits exponential scaling with respect to the dimensions and does not take on the curse of dimensions.

1.4. Constraining Correlations

Until here, our efforts were focused on obtaining data-sparse representations of interactions, Hamiltonians, potentials. This is to say, the curse of dimension was taken as is; the goal was to improve upon the power-two scaling. Practically speaking, the situations covered are such that a state can be handled nicely, whereas an operator tops the limit of computational resources.

In this section we will take on the curse of dimensions and focus our attention on states, rather than operators. By that we mean that knowledge about the structure of the state (for example a wave-function during time propagation) is used to limit the amount of computational resources required to store and handle it. The most straight-forward case is the constraining of a Hilbert space onto a subspace. For example, symmetries of the Hamiltonian dictate values of conserved quantities if these are known initially.

This section focuses on quantum mechanics on a Hilbert space. The aforementioned use of a conserved charge is a linear constraint in this setting. We restrict the Hilbert space to a subspace (which is of course another Hilbert space). When dealing with

high-dimensional (in other words many-body) quantum mechanics, there are however more important constraints, which can be employed. The reasoning here is that “Hilbert space is huge” and we do not need all of it. “Most of Hilbert space” consists of “weirdly entangled/correlated” states, which will never be attained in real-life applications. The uncorrelated states, product states, only make up an exponentially small (by this we mean for a Hilbert space of dimension n^d , uncorrelated states can be described using dn numbers) fraction of Hilbert space. As we will see, once these notions are put into a less hand-waving framework, there are many possibilities to modulate between uncorrelated and fully-correlated states, where the states of interest only make up an exponentially small fraction of Hilbert space and therefore can be tackled computationally.

Constraints on the correlation beat the curse of dimensions; this however, is not achieved by choosing a subspace. Rather, we restrict our attention to subsets (not necessarily manifolds; however one can usually think of the subsets as being sub-manifolds) of the Hilbert space. As an example, the space of product states (rank-1 tensors) is *not* a linear subspace of $\mathbb{C}^n \otimes \dots \otimes \mathbb{C}^n$.

We will start with exactly these states, commonly referred to as *Hartree type approximations*; generalizing the concept one arrives at *multiconfiguration time dependent Hartree (Fock)* methods. Specializing on local correlations in an orderable configuration (i. e. spin chain) one arrives at *matrix product states*, which are generalized to *tensor networks*.

1.4.1. Distinguishability

To set the stage, we introduce some notation related to the fact that we are dealing with three distinct cases. Quantum mechanics for many-body problems needs to address the problem of distinguishability of the particles. Let the Hilbert space for d particles (or sites) be given by

$$\mathcal{H} = \mathcal{H}_1 \otimes \dots \otimes \mathcal{H}_d.$$

This is the configuration space in case the particles are *distinguishable*. If they are not, as for example electrons in a molecule are, by the *Pauli principle*, the space needs to be constrained to either symmetric or antisymmetric tensors, depending on spin symmetry and the distinction between fermions and bosons. In this case, we have $\mathcal{H}_i = \mathcal{H}_j = \mathcal{H}_1$ and we denote projectors onto the space of (anti-)symmetric states by P_{\pm} ; they are given by

$$(P_{\pm}\psi)_{i_1\dots i_d} = \frac{1}{\sqrt{d!}} \sum_{\pi} (\pm)^{\pi} \psi_{i_{\pi(1)}\dots i_{\pi(d)}},$$

where the sum runs over all permutations $\pi : \{1, \dots, d\} \rightarrow \{1, \dots, d\}$ and $(-)^{\pi}$ denotes the sign of the permutation. The spaces of (anti-)symmetric wave-functions are of complex dimensions

$$\begin{aligned}\dim(P_+\mathcal{H}) &= \binom{\dim \mathcal{H}_1 + d - 1}{\dim \mathcal{H}_1 - 1}, \\ \dim(P_-\mathcal{H}) &= \binom{\dim \mathcal{H}_1}{d}.\end{aligned}$$

We remark that $\dim(P_+\mathcal{H}) + \dim(P_-\mathcal{H}) = \dim \mathcal{H}$ only holds for $d = 2$. In other words, higher order tensors *cannot* be decomposed into totally symmetric and anti-symmetric parts. The curse of dimensions is present only in the setting $n \gg d$.

We will refer to this threefold distinction between distinguishable, fermionic, and bosonic states as exchange symmetry. Note that the constraints of fermionic or bosonic states are still linear ones; that is, $P_{\pm}\mathcal{H}$ are subspaces of \mathcal{H} .

1.4.2. The Dirac-Frenkel Variational Principle

As we are constraining the configuration space, it is a priori not clear how the Schrödinger equation is to be implemented on the subset of admissible states. First of all, when discretizing an (infinite-dimensional) Hilbert space to a finite-dimensional subspace spanned by $\{e_i : i = 1, \dots, n\}$, we implement time propagation by

$$i \frac{\partial}{\partial t} \psi_i = \sum_{j=1}^n H_{ij} \phi_j, \quad \psi_i = \langle e_i | \psi \rangle, \quad H_{ij} = \langle e_i | H | e_j \rangle.$$

That way, the state ψ remains constrained for all times. This equation is equivalent to requiring

$$\langle \delta\psi | H - i \frac{\partial}{\partial t} | \psi \rangle = 0$$

for all variations $\delta\psi$ of (tangents $\delta\psi$ at) ψ . The last equation generalizes immediately to the case where ψ is not constrained to a subspace; it is known as the *Dirac-Frenkel variational principle* [28, 29]. For a further review of this principle we refer the reader to [5].

1.4.3. Hartree Type Approximations and CP States

Applying a Hartree type approximation corresponds to constraining the basis to be a *product state*, where what is meant by that depends on exchange symmetry. In the case of distinguishable particles, we define the space of Hartree functions as

$$H_D(\mathcal{H}_1, d) = \left\{ \psi \in \mathcal{H}_1^{\otimes d}, \text{rank}(\psi) = 1 \right\},$$

which is the space of tensor product states. The subscript D stands for “distinguishable.” These states are totally uncorrelated, i. e. for observables A and B located on disjoint subsets of sites, we have $\langle \psi | AB | \psi \rangle = \langle \psi | A | \psi \rangle \langle \psi | B | \psi \rangle$.

For fermionic/bosonic states, we define

$$H_{\pm}(\mathcal{H}_1, d) = P_{\pm}H_D(\mathcal{H}_1, d).$$

Restricting the configuration space to $H_{-}(\mathcal{H}_1, d)$ is usually referred to as (*time-dependent*) *Hartree-Fock*. In practice, Hartree-Fock is used to give upper bounds on eigenenergies of multi-electron systems (molecules). In that case, $\mathcal{H}_1 = L^2(\mathbb{R}^3)$ and one needs to use a discretization, i. e. a finite-dimensional subspace $\mathcal{H}_{1D} \subset \mathcal{H}_1$. One is then constrained to $H_{-}(\mathcal{H}_{1D}, d) \subset H_{-}(\mathcal{H}_1, d)$. Notice, however, that this is a fundamentally different form of constraint. By increasing the dimension \mathcal{H}_{1D} we can approximate any vector $\phi \in \mathcal{H}_1$ arbitrarily well; but we *cannot* approximate any vector in $P_{-}\mathcal{H}_1^{\otimes d}$ arbitrarily well within $H_{-}(\mathcal{H}_1, d)$.

The vectors in $H_{-}(\mathcal{H}_1, d)$ are usually referred to as *Slater determinants*. This follows from the fact that inserting a rank-one tensor into P_{-} immediately gives us

$$\psi \in H_{-}(\mathcal{H}_1, d) \Rightarrow \psi_{i_1 \dots i_d} = \frac{1}{\sqrt{d!}} \det \begin{pmatrix} \psi_{i_1}^{(1)} & \psi_{i_1}^{(2)} & \dots \\ \psi_{i_2}^{(1)} & \psi_{i_2}^{(2)} & \dots \\ \vdots & \vdots & \ddots \end{pmatrix}$$

for a set of vectors $\psi^{(f)} \in \mathcal{H}_1$. These vectors can always be chosen orthonormal, as $\det(AB) = \det(A)\det(B)$. Notice that as opposed to H_D the spaces H_{\pm} contain tensors of ranks higher than one. Eckert et al [30] introduce fermionic and bosonic Slater ranks of (anti-)symmetric tensors, which generalize tensor ranks in the sense that H_{\pm} , as well as H_D , consist of rank-1 tensors.

Two-Particle Correlations Let us elaborate on the correlation that can be implemented in Hartree-Fock in the case $H_{-}(\mathbb{C}^n, 2)$, where the states are matrices. Let ψ be such a Hartree-Fock vector

$$\psi_{ij} = \frac{1}{\sqrt{2}} \left(\psi_i^{(1)} \psi_j^{(2)} - \psi_i^{(2)} \psi_j^{(1)} \right), \quad \langle \psi^{(\alpha)} | \psi^{(\beta)} \rangle = \delta_{\alpha\beta}.$$

We introduce the covariance for a one-particle observable S as

$$\text{cov}_S = \langle \psi | S \otimes S | \psi \rangle - \langle \psi | S \otimes \text{Id} | \psi \rangle \langle \psi | \text{Id} \otimes S | \psi \rangle.$$

One quickly checks that

$$\text{cov}_S = -\frac{1}{4} \left(\langle \psi^{(1)} | S | \psi^{(1)} \rangle - \langle \psi^{(2)} | S | \psi^{(2)} \rangle \right)^2 - \left| \langle \psi^{(1)} | S | \psi^{(2)} \rangle \right|^2 < 0.$$

Therefore, Hartree-Fock states only allow for *anti-correlations*. One might be tempted to believe that this is due to the antisymmetry of the wave-function; however, this is not the case. Two electrons in a molecule may well be correlated positively due to the influence of the Coulomb interaction. To show this, we need to distinguish two cases: $n \leq 3$ and $n > 3$. It is not hard to see that $P_{-}\mathbb{C}^{2 \times 2} = H_{-}(\mathbb{C}^2, 2)$. This extends to $n = 3$, which can be seen from the cross-product, that establishes a

correspondence between 3-forms and vectors in \mathbb{C}^3 . In other words, every tensor in $P\text{-}\mathbb{C}^{3 \times 3}$ corresponds to a vector in \mathbb{C}^3 ; this vector can be written as a cross product, a Hartree-Fock state.

Furthermore, in the case $n = 2$ we can simplify the above equation by choosing a basis of \mathbb{C}^2 that diagonalizes S ; without loss of generality $\psi_2^{(1)} = 0$, $\psi_1^{(2)} = 0$, and the last term drops. Calculating the variances

$$\begin{aligned}\sigma_1^2 &= \langle \psi | (S - \langle \psi | S \otimes \text{Id} | \psi \rangle)^2 \otimes \text{Id} | \psi \rangle, \\ \sigma_2^2 &= \langle \psi | \text{Id} \otimes (S - \langle \psi | \text{Id} \otimes S | \psi \rangle)^2 | \psi \rangle,\end{aligned}$$

we obtain

$$\sigma_1 = \sigma_2 = \sigma, \quad \text{cov}_S = -\sigma^2.$$

The correlation $\text{cor}_S = \text{cov}_S / \sigma_1 \sigma_2$ between both particles is -1 (for every observable where $\sigma \neq 0$).

The general case for $n > 3$ allows for positive covariance, as can be seen from the example

$$\psi = \frac{1}{2} \begin{pmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \end{pmatrix}, \quad S = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix},$$

for which one readily calculates

$$\text{cov}_S = 1, \quad \sigma = 1 \Rightarrow \text{cor}_S = 1.$$

CP States One can generalize this concept in a straight-forward manner by constraining the space of states to tensors of $\text{rank}(\psi) \leq R$. We call this method *CP states* following the canonical decomposition introduced in Section 1.3.2. In the literature, such approximations appear as *configuration interaction*. This class of approximations has a nice geometrical visualization [6] given in Fig. 1.6. We imagine an order-2 tensor as a shape in \mathbb{R}^2 and each rank in a low-rank approximation as a box, i. e. as an outer product of two vectors. The figure is nothing more than a visualization to aid ones imagination; it is not to be understood as an exact representation of mathematical quantities.

Beylkin and Mohlenkamp [31] start from CP states and develop an efficient numerical paradigm; additionally they show for the d -electron Schrödinger Hamiltonian that a representation with rank growing logarithmically in d can be used to approximate the Schrödinger equation arbitrarily well. This effectively beats the curse of dimensions.

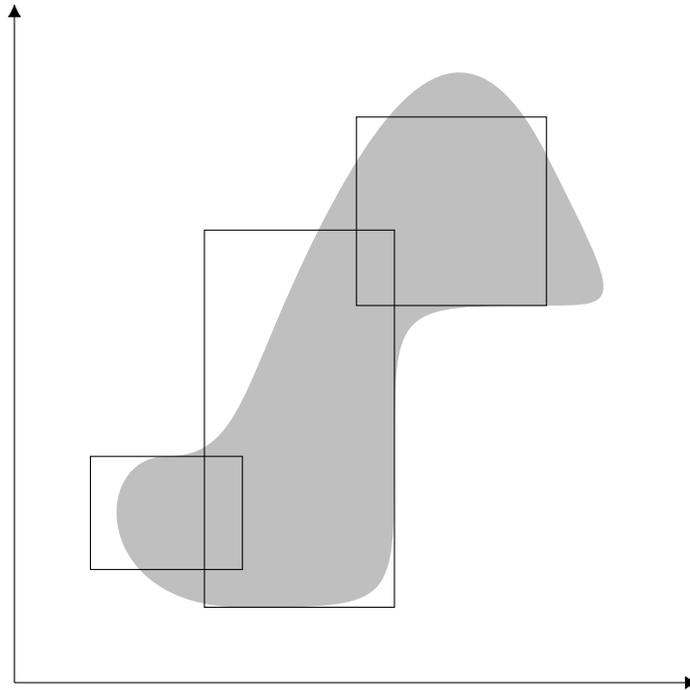


Figure 1.6.: Geometrical interpretation of a CP state [6]. The gray area represents the tensor to be approximated; each of the black boxes is a product state, i. e. “one rank” in the decomposition. The rank of the approximation is 3 in this case.

1.4.4. Tucker States

In the last section we applied the canonical decomposition to limit the configuration space to contain only low-rank tensors. As pointed out earlier, there is a more straight-forward generalization of rank from matrices to tensors, the HOSVD or Tucker representation. Limiting the f -rank of the tensor ψ instead of the rank leads us to the *multiconfiguration time dependent Hartree* (MCTDH) method [5] and its fermionic counterpart MCTDHF [6, 32]. MCTDH is defined by the space

$$M_D(\mathcal{H}_1, d, k) = \left\{ \psi \in \mathcal{H}_1^{\otimes d}, \text{rank}_n(\psi) = k_n \right\}, \quad k \in \mathbb{N}^d,$$

with the Fock and bosonic analogues given by

$$M_{\pm}(\mathcal{H}_1, d, k) = P_{\pm} M_D(\mathcal{H}_1, d, (k, \dots, k)), \quad k \in \mathbb{N}.$$

As is the case for CP states, MCTDH(F) can be understood as modulating between Hartree(-Fock) and the full Hilbert space. By increasing the f -ranks from 1 to $\dim \mathcal{H}_1$, one is able to move from Hartree-Fock approximation to a fully-correlated representation. The interpretation of the constraints implemented by a low-rank MCTDH(F) approximation can be visualized best in a different manner from POTFIT. Whereas in POTFIT the key idea is an oversampled potential, one represented in a too fine basis, for MCTDH(F) the key lies in the realization that the basis vectors $a^{(f)}$ are dynamical quantities. Time propagation can be modelled well in the approximation if for all times t there are projectors of low rank $P_i(t)$ such that

$$\psi(t) \approx (\text{Id} \otimes \dots \otimes P_i(t) \otimes \dots \otimes \text{Id}) \psi(t),$$

where $P_i(t)$ acts on the i 'th degree of freedom. It is important to note that within the space

$$(P_1(t) \otimes P_2(t) \otimes \dots \otimes P_d(t)) P_{\pm} \mathcal{H}_1^{\otimes d}$$

any form of correlation can be represented exactly. One can visualize these subspaces, for example, if the electrons remain well localized during time propagation; in this case the $P_i(t)$ project onto $L^2(B_i(t))$, where the $B_i(t) \subset \mathbb{R}^3$ “contain most of $\psi(t)$.”

Alternatively, one can interpret any Tucker decomposition with f -ranks r_f as a CP with rank $r_1 \dots r_d$ with additional constraints on the factor matrices. Then, the interpretation of MCTDH(F) aligns with CP states; viewing entanglement as a complex high-dimensional geometry, we approximate it through a sum of “boxes.” However, without this viewpoint, we can extend the previous visualization in a crude manner (crude, as for a rank-2 tensor a Tucker core tensor can always be diagonalized; in the case of two degrees of freedom CP States and Tucker States are equivalent). This is presented in Fig. 1.7. Instead of approximating the tensor as a sum of boxes, we transform each coordinate to a basis, “well-suited for the tensor” and give a full tensor within this basis.

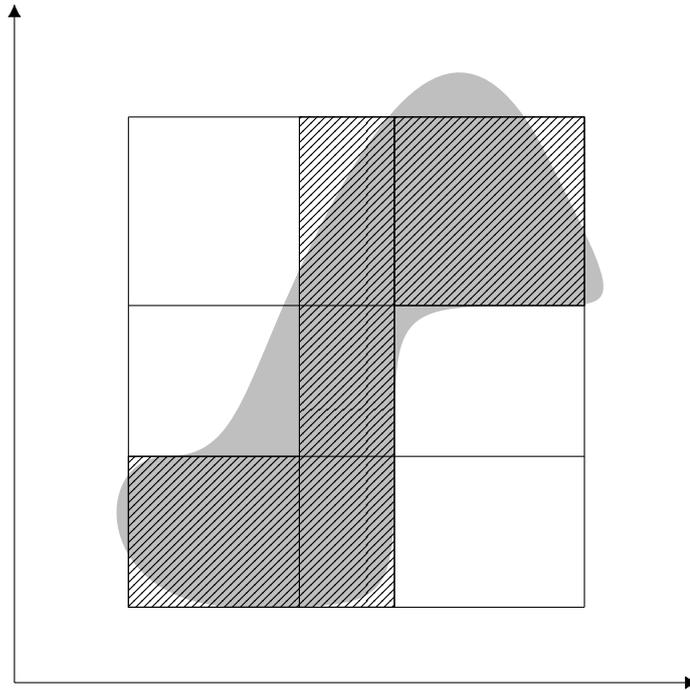


Figure 1.7.: Geometrical interpretation of a Tucker state. The full tensor is given in gray; black lines represent Tucker basis and filled out rectangles the core tensor in this basis; the approximation given corresponds to f -rank 3 in both degrees of freedom.

For an example of a real-world application of MCTDHF, we refer the reader to [33], where convergence for two systems is shown and compared to Hartree-Fock, which yields qualitatively different results.

1.4.5. Matrix Product States

The Hartree type approximations and their derivatives do not assume a certain ordering of the degrees of freedom in the sense that certain degrees of freedom interact differently with certain others. In fact, most successfully used are the fermionic/bosonic versions of the approximations, which inherently assume indistinguishability. Let us now focus on a system, whose degrees of freedom can in fact be aligned linearly, e. g. a spin chain, or ring. The term “degree of freedom” is not a precise one; we mean by it that the Hilbert space of the system is decomposed (in a physically motivated way) into a tensor product of factor spaces; each factor space is called a degree of freedom, a particle, or a site. Obviously, the particles are now distinguishable and we will not project onto fermionic or bosonic spaces.

Assuming a local interaction, it stands to reason that the configuration space needs only to include correlations that are stronger between closer separated particles and weaker for far distances. In fact, Hastings [34] proves this fact for local gapped Hamiltonians rigorously. The ground state of local gapped Hamiltonians has a correlation length; that is, correlations decay exponentially. (We will make these notions rigorous in a moment.) These local gapped Hamiltonians build the context in which *matrix product states* (MPS) were first introduced by Affleck, Kennedy, Lieb and Tasaki [7], named *AKLT states* after the authors initials, and as *finitely correlated states* by Fannes, Nachtergaele and Werner [8]. They were later seen to be the states arising in *density matrix renormalization group theory* (see [35] and references therein). MPS have also been introduced as *tensor trains* (TT) [36].

Definitions This section reports results from [9] following the notation used there. Assume all particles live in the same Hilbert space; generalizations are straightforward. Denote this Hilbert space by \mathcal{H}_1 and a basis by $\{e_1, \dots, e_n\}$. A state $\psi \in \mathcal{H}_1^{\otimes d}$ is an MPS with *bond dimension* D if it can be written as

$$\psi \propto \sum_{i_1 \dots i_d} \text{tr} \left[A_{i_1}^{(1)} A_{i_2}^{(2)} \dots A_{i_d}^{(d)} \right] e_{i_1} \otimes e_{i_2} \otimes \dots \otimes e_{i_d},$$

for $A_i^{(f)} \in \mathbb{C}^{D_f \times D_{f+1}}$ and we have $D_f \leq D$ for all f . The components of the vector are given by matrix products, giving ψ the name MPS. Since it is inconvenient to absorb the normalization into the matrices, we will throughout this text refer to ψ as the normalized state vector arising from the MPS representation $A_i^{(f)}$.

For a matrix $S \in \mathbb{C}^{n \times n}$ we introduce the operators

$$E_S^{(f)} : \mathbb{C}^{D_{f+1} \times D_{f+1}} \rightarrow \mathbb{C}^{D_f \times D_f},$$

$$\rho \mapsto \sum_{ij} S_{ij} A_j^{(f)} \rho A_i^{(f)\dagger}.$$

We quickly verify that the adjoint of this map under $\langle A, B \rangle = \text{tr} [A^\dagger B]$ is given by

$$E_S^{(f)\dagger} : \mathbb{C}^{D_f \times D_f} \rightarrow \mathbb{C}^{D_{f+1} \times D_{f+1}},$$

$$\rho \mapsto \sum_{ij} S_{ij} A_i^{(f)\dagger} \rho A_j^{(f)}.$$

By $E^{(f)} = E_{\text{Id}}^{(f)}$ we denote the *transfer matrix*. Note that the transfer matrix as well as its adjoint are manifestly positive. In fact, one checks easily that they are *completely positive*.⁷

Canonical Form Obviously, one can transform $A_i^{(f)'} = A_i^{(f)} X$, $A_i^{(f+1)'} = X^{-1} A_i^{(f+1)}$ for every non-singular X , implying this representation is not unique. However, following [9], any MPS representation can be brought to *canonical form*. We now distinguish two cases; either the MPS is called *open*, given by $D_1 = D_{f+1} = 1$, where the first and last particle are not considered to be neighbours (strongly correlated). Or we call an MPS *translationally invariant* (TI) if $D_i = D$ and ψ is invariant under cyclic permutations of the indices.

Any open MPS can be brought to canonical form [9]

$$E^{(f)}(\text{Id}) = \text{Id},$$

$$E^{(f)\dagger}(\Lambda^{(f-1)}) = \Lambda^{(f)}, \tag{1.13}$$

where the matrices $\Lambda^{(f)}$ are density matrices (positive and trace one), diagonal, and of full rank. Furthermore, we have $\Lambda^{(0)} = \Lambda^{(d)} = 1$. The canonical form is unique up to permutations and unitary transformations on degenerate eigenspaces of the $\Lambda^{(f)}$. Notice that this canonical form corresponds to choosing $E^{(f)}$ to be unital and $E^{(f)\dagger}$ trace-preserving.⁸ In other words, the adjoints of the transfer matrices are *quantum channels*.⁹ This definition of canonical form is usually referred to as *right-canonical* [35]. One can equally well require the transfer matrices to be quantum channels and arrive at *left-canonical* MPS.

⁷A map $A : \mathbb{C}^{n \times n} \rightarrow \mathbb{C}^{m \times m}$ is called positive if the image of a positive matrix ρ ($\langle \psi, \rho \psi \rangle \geq 0 \forall \psi$) is again positive. We call A completely positive if the map $\text{Id}_{k \times k} \otimes A : \mathbb{C}^{k \times k} \otimes \mathbb{C}^{n \times n} \rightarrow \mathbb{C}^{k \times k} \otimes \mathbb{C}^{m \times m}$,

$M \otimes \rho \mapsto M \otimes A(\rho)$ is positive for every k . For further details see Appendix A.1.

⁸A map is called unital if it maps the identity to the identity. This immediately implies that its adjoint is trace-preserving.

⁹A quantum channel is a completely positive trace-preserving map.

As we will see later, the canonical representation of any tensor ψ can be obtained from the SVD of a matricization of ψ . This implies that any ψ can be represented as an MPS if only the bond dimension is chosen high enough, where the bound $D \leq n^{d/2}$ holds [37]. This means, in general D grows exponentially reproducing the curse of dimensions. In practice, we want to apply MPS in situations where D grows only polynomial with decreasing truncation error.

Following [9], for TI MPS, the canonical form is given by site-independent matrices $A_i^{(f)} = A_i$ that are block-diagonal

$$A_i = \begin{pmatrix} \lambda_1 A_{i1} & 0 & \dots \\ 0 & \lambda_2 A_{i2} & \dots \\ \vdots & \vdots & \ddots \end{pmatrix},$$

satisfying

$$\begin{aligned} \sum_i A_{ij} A_{ij}^\dagger &= \text{Id}, \\ \sum_i A_{ij}^\dagger \Lambda_j A_{ij} &= \Lambda_j, \end{aligned}$$

where the matrices Λ_j are of full rank and diagonal with positive entries. Furthermore we have

$$\sum_i A_{ij} X A_{ij}^\dagger = X \Rightarrow X \propto \text{Id}.$$

[9] gives necessary conditions, under which this canonical representation is unique. In other words, any TI MPS is a sum of so called *ergodic* TI MPS, where we call a TI MPS ergodic, if the transfer matrix has only the identity as fixed point.

Some Examples The GHZ state [38] defined, for $n = 2$, as the superposition of all particles in one state $|0\rangle$ and all particles in the other state $|1\rangle$

$$\psi = \frac{1}{\sqrt{2}} (|0\dots 0\rangle + |1\dots 1\rangle)$$

can be represented as an MPS state with bond dimension two

$$A_0^{(f)} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \quad A_1^{(f)} = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}.$$

This representation satisfies, aside from using the trace instead of vectors $A^{(1)}$ and $A^{(f)}$, the conditions of the canonical form for an open MPS; however it is not in canonical form considered as a TI MPS as all diagonal matrices are fixed points

of the transfer matrix. However both summands $|0\dots 0\rangle$ and $|1\dots 1\rangle$ are trivially canonical TI MPS.

Notice that any two-particle reduced density matrix is given by

$$\rho = \frac{1}{2} (|00\rangle\langle 00| + |11\rangle\langle 11|);$$

therefore, the GHZ state does not limit correlations based on positions in the chain; this, obviously, cannot happen as it is totally symmetric. We will come back to the question of two-particle correlations in the following section.

More generally, any tensor of rank D can be represented as an MPS with bond dimension D [36]. Suppose we have the rank decomposition

$$\psi_{i_1\dots i_d} = \sum_{k=1}^D \psi_{ki_1}^{(1)} \dots \psi_{ki_d}^{(d)}.$$

Then we can trivially represent ψ using

$$\left(A_i^{(f)}\right)_{\alpha\beta} = \delta_{\alpha\beta} \psi_{\alpha i}^{(f)}.$$

As the GHZ state, these states do, in general, not exhibit any correlation constraining based on relative location.

One-Particle Observables We will later study correlation functions of MPS; to this end, we note that the expectation value of a product of one-particle observables can be obtained as follows [9]

$$\langle \psi | S^{(1)} \otimes \dots \otimes S^{(d)} | \psi \rangle = \text{tr} \left[E_{S^{(1)}}^{(1)} \dots E_{S^{(d)}}^{(d)} \right] / \text{tr} \left[E^{(1)} \dots E^{(d)} \right],$$

where the product of $E_S^{(f)}$ is to be understood as composition. This follows from the fact that

$$\sum_{ij} S_{ij} A_j^{(f)} \otimes A_i^{(f)*},$$

is a matrix representation of $E_S^{(f)}$ and we can use $\text{tr} [A \otimes B] = \text{tr} [A] \text{tr} [B]$.

Notice that for an open MPS in canonical form the composition of all $E_S^{(f)}$ is a map from $\mathbb{C}^{1 \times 1}$ to $\mathbb{C}^{1 \times 1}$ and we have

$$\langle \psi | S^{(1)} \otimes \dots \otimes S^{(d)} | \psi \rangle = E_{S^{(1)}}^{(1)} \dots E_{S^{(d)}}^{(d)} (1); \quad (1.14)$$

in particular, it is normalized.

Two-Particle Correlation Functions We will now investigate the correlation implemented in MPS. It is a well known fact that correlation functions decay exponentially [8, 9, 39]; that is, MPS can represent states well that have a correlation length. Let us denote the two-particle correlation function for the observable S by

$$\text{cov}_S(m, n) = \langle \psi | S^{(m)} S^{(n)} | \psi \rangle - \langle \psi | S^{(m)} | \psi \rangle \langle \psi | S^{(n)} | \psi \rangle,$$

where $S^{(n)} = \text{Id} \otimes \cdots \otimes \text{Id} \otimes S \otimes \text{Id} \otimes \cdots \otimes \text{Id}$ with S sitting at the n 'th spot.

From here on we follow [39] and present the calculation without requiring translational invariance. To investigate this correlation function for an open MPS in right-canonical form, we assume $m < n$ and denote by

$$\begin{aligned} E^A &= E^{(1)} \dots E^{(m-1)}, \\ E_S^1 &= E_S^{(m)}, \quad E^1 = E^{(m)}, \\ E^B &= E^{(m+1)} \dots E^{(n-1)}, \\ E_S^2 &= E_S^{(n)}, \quad E^2 = E^{(n)}, \\ E^C &= E^{(n+1)} \dots E^{(d)}. \end{aligned}$$

Employing (1.14), we find the correlation function

$$\begin{aligned} \text{cov}_S(m, n) &= (E^A E_S^1 E^B E_S^2 E^C (1)) (E^A E^1 E^B E^2 E^C (1)) \\ &\quad - (E^A E_S^1 E^B E^2 E^C (1)) (E^A E^1 E^B E_S^2 E^C (1)). \end{aligned} \quad (1.15)$$

The goal now is to relate the operator E^B , describing the entanglement between the two sites in some manner to the actual correlation. To this end, we apply the singular value decomposition to $E^B(\rho) = \sum_{\alpha} \sigma_{\alpha} u_{\alpha} \langle v_{\alpha}, \rho \rangle$ with singular values $\sigma_{\alpha} \geq \sigma_{\alpha+1}$ and normed singular vectors u_{α}, v_{α} . We obtain, after the calculation presented in Appendix A.2,

$$|\text{cov}_S(m, n)| \leq C_D \|S\|_{\text{op}}^2 \frac{\sigma_2}{\sigma_1},$$

with a constant C_D , depending on the bond dimension D , and the operator norm $\|S\|_{\text{op}}$ of the observable.

Let us take a look at an example presented in [39]. Suppose we have a translationally invariant situation $A_i^{(f)} = A_i^{(f+1)}$ for $1 < f < d - 1$. Suppose further the transfer matrix for any site not located at the ends is diagonalizable with eigenvalues $|\lambda_1| > |\lambda_2| \geq |\lambda_3| \dots$. Then the eigenvalues of E^B are given by $\sigma_1 = \lambda_1^{n-m-2}$, $|\sigma_2| = |\lambda_2|^{n-m-2}$ and we have

$$|\text{cov}_S(m, n)| \leq C_D \|S\|_{\text{op}}^2 \left| \frac{\lambda_2}{\lambda_1} \right|^{|m-n|-2} \propto \exp \left[-\frac{|m-n|}{\xi} \right],$$

which is exponential decay with correlation length $\xi = \log(|\lambda_1/\lambda_2|)^{-1}$. As expected ξ diverges for $|\lambda_2| \rightarrow |\lambda_1|$, and goes to zero for $\lambda_2 \rightarrow 0$.

This example appears to be quite special; however, we reason that most MPS states obey an exponential correlation decay law of some sort. The probability, when picking an MPS state randomly, is hugely in favour of picking one that obeys an exponential correlation decay law for one particle observables. To make this hand-waving more clear, we conduct a crude numerical experiment based on the bound obtained above.

Given D and n , we pick quantum channels $E(\rho) = \sum_{i=1}^n A_i^\dagger \rho A_i$ by random (following the algorithm outlined in [40]) and calculate the leading two singular values of a product of l of these. This means, we model σ_2/σ_1 for $|m-n|-2=l$. We restrict ourselves to quantum channels, i.e. canonically represented MPS, because the bound is only valid under these circumstances. Furthermore, by eliminating the gauge freedom, we indeed achieve a parametrization of matrix product states, and not representations of these. We plot the values of $\log[\sigma_2/\sigma_1]$ and the means and standard deviations thereof. Our results are presented in Fig. 1.8. We see that the expectation value of $\log[\sigma_2/\sigma_1]$ behaves linearly with l . Although at every l we are able to find states that have two nearly identical highest singular values, the bulk of MPS states will obey an exponential correlation law. It is worth noting that these results rely on our, quite crude, bound above and could probably be improved.

Relation to Area Law of Entropy The last section discussed one-particle correlation functions as a measurement of entanglement. This, however, is only one facet of the vastness of Hilbert space. In the context of MPS, the entanglement entropy provides a well-suited measure of entanglement; by that we mean, there is a correspondence between states describable as MPS and states obeying certain bounds on entanglement entropy.

We define the von-Neumann entropy of a density matrix ρ as

$$S(\rho) = -\text{tr}[\rho \ln \rho].$$

Let us now divide the system into two subsystems: $A = \{1, \dots, k\}$, $B = \{1, \dots, d\} \setminus A$. Consider the pure state $\rho = |\psi\rangle\langle\psi|$, $\psi \in (\mathbb{C}^n)^{\otimes d}$ and the reduced density matrices $\rho_{A/B} = \text{tr}_{B/A}(\rho)$, where $\text{tr}_{A/B}$ denotes the partial trace over A resp. B . To make the motivation for the above definition more clear, we take a closer look at $S(\rho_{A/B})$. First of all, by the Schmidt decomposition (SVD) we can write ψ as

$$\psi = \sum_i c_i \phi_i^{(A)} \otimes \phi_i^{(B)},$$

where $\phi_i^{(A/B)}$ live on the factor spaces corresponding to the two parts. We readily calculate

$$\rho_{A/B} = \sum_i |c_i|^2 |\phi_i^{(A/B)}\rangle\langle\phi_i^{(A/B)}| \Rightarrow S(\rho_{A/B}) = -\sum_i |c_i|^2 \ln(|c_i|^2).$$

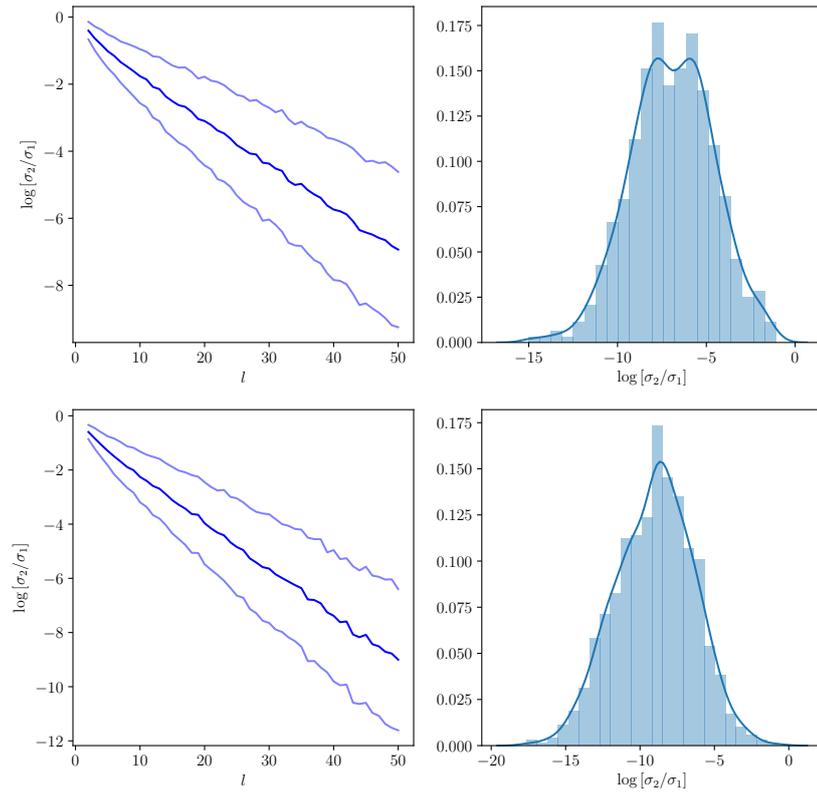


Figure 1.8.: Quotient of first and second singular value of a product of l random quantum channels. Figures on top correspond to $D = 2, n = 2$; bottom figures correspond to $D = 2, n = 5$. Left side gives mean and standard deviation with respect to l . Right side figures show the distribution of $\log[\sigma_2/\sigma_1]$ for $l = 50$.

The entropy does not depend on which subsystem is chosen; furthermore it is the natural extension of the classical Gibbs entropy, where $S = 0$ for a product (uncorrelated) state.

Assume now $k \ll d$. The maximal value of the entropy is obtained if all c_i are of the same magnitude. Therefore, we have $|c_i|^2 = n^{-k}$ with the entropy

$$S_{\max} = k \ln(n),$$

i. e. the entropy grows linearly with the size (volume) of the subsystem A .

MPS are closely related to an *area law of entanglement* [34, 41, 42], i. e. the entanglement entropy grows linearly with the area of the boundary of A . In one dimension this boundary area is constant. That is, for 1D problems, the entanglement entropy is bounded by a constant. If the d sites were aligned in a plane $\sqrt{d} \times \sqrt{d}$, the boundary area grows linearly with \sqrt{k} and an area law would imply that the entropy scales at most like \sqrt{k} .

The relation between MPS and area laws is established by Brandão [42]. This paper proves an area law of entanglement for a system with finite correlation length under certain circumstances. We have seen that MPS describe systems well that obey exponential decay. In fact, the same paper proves (again in a special case) that any system with finite correlation length can be approximated well by an MPS. This tightens the correspondence of exponential decay of correlations and MPS, as well as it relates these correlations to area laws of entanglement.

Hastings [34] proves that any local gapped Hamiltonian with finite interaction strength has a ground state that obeys an area law of entanglement. Furthermore, any such Hamiltonian can be solved approximately in the framework of matrix product states [41], strengthening the correspondence between exponential decay of correlations and an area law of entanglement.

Truncation to MPS As stated above, any tensor $\in (C^n)^{\otimes d}$ can be represented exactly with bond dimension $D \leq n^{d/2}$. This follows in a constructive way from the TT-SVD algorithm presented in [35–37]. We outline this algorithm here and refer the reader to the above citations for greater detail.

Suppose ψ is of shape $n_1 \times \cdots \times n_d$. Start by reshaping ψ to $\psi^{(1)}$ of shape $n_1 \times \left(\prod_{i=2}^d n_i\right)$; this is a matrix, whose rows are labelled by i_1 and columns are labelled by the multi-index $(i_2 \dots i_d)$. Compute the SVD of this matrix to arrive at $\psi^{(1)} = U^{(1)} \sigma^{(1)} V^{(1)\dagger}$, where $U^{(f)}$ is of shape $n_1 \times D_2$, $V^{(1)}$ is of shape $\left(\prod_{i=2}^d n_i\right) \times D_2$ and D_2 is the rank of $\psi^{(1)}$. The rows of $U^{(1)}$ now form the vectors $A_{i_1}^{(1)} \in \mathbb{C}^{1 \times D_2}$ and we have

$$\psi_{i_1, i_2, \dots, i_d} = \sum_{\alpha} \left(A_{i_1}^{(1)}\right)_{1\alpha} \sigma_{\alpha}^{(1)} V_{\alpha, (i_2 \dots i_d)}^{(1)\dagger}.$$

We now iterate this procedure, reshaping $\sigma^{(1)} V^{(1)\dagger}$ to $\psi^{(2)}$ of shape $(D_2 n_2) \times \left(\prod_{i=3}^d n_i\right)$. This gives us, recursively going on,

$$\begin{aligned}
 \psi_{i_1 i_2 \dots i_d} &= \sum_{\alpha} \left(A_{i_1}^{(1)} \right)_{1\alpha} \psi_{(\alpha i_2), (i_3 \dots i_d)}^{(2)} \\
 &= \sum_{\alpha\beta} \left(A_{i_1}^{(1)} \right)_{1\alpha} U_{(\alpha i_2), \beta}^{(2)} \psi_{(\beta i_3), (i_4 \dots i_d)}^{(3)} \\
 &= \sum_{\alpha\beta} \left(A_{i_1}^{(1)} \right)_{1\alpha} \left(A_{i_2}^{(2)} \right)_{\alpha\beta} U_{(\beta i_3), \gamma}^{(3)} \psi_{(\gamma i_4), (i_5 \dots i_d)}^{(4)},
 \end{aligned}$$

and finally an MPS representation of ψ . In summary, at each step, we have

$$\begin{aligned}
 \left(A_{i_f}^{(f)} \right)_{\alpha\beta} &= U_{(\alpha i_f), \beta}^{(f)}, \\
 \psi_{(\beta i_{f+1}), (i_{f+2} \dots i_d)}^{(f+1)} &= \sigma_{\beta}^{(f)} V_{\beta, (i_{f+1} \dots i_d)}^{(f)\dagger},
 \end{aligned}$$

for an SVD $\psi^{(f)} = U^{(f)} \sigma^{(f)} V^{(f)\dagger}$. By the unitarity of $U^{(f)}$, we immediately have $E^{(f)\dagger}(\text{Id}) = U^{(f)\dagger} U^{(f)} = \text{Id}$. Analogously, the unitarity of $V^{(f)}$ gives us

$$E^{(f)} \left(\left(\sigma^{(f)} \right)^2 \right) = \left(\sigma^{(f-1)} \right)^2,$$

arriving at the left-canonical form. In fact, the squares of the singular values $\sigma^{(f)}$ correspond to eigenvalues of reduced density matrices $\text{tr}_{1 \dots (f-1)} |\psi\rangle\langle\psi|$ [9].

Analogously to the above algorithm, one arrives at a right-canonical form by starting the unfolding from the right.

We are interested mostly, however, not in exact representations, but in approximations of tensors. We can straight-forwardly find an approximation of a given tensor by truncating the SVDs at any stage of the above algorithm; denote the resulting tensor by ψ_{trunc} . Oseledets [36] shows that, if the truncation error in the SVD $\psi^{(f)} \approx U^{(f)} \sigma^{(f)} V^{(f)\dagger}$ is bounded by ϵ_k , the truncation error of the full tensor is bounded by

$$\|\psi - \psi_{\text{trunc}}\|_F^2 \leq \sum_{f=1}^{d-1} \epsilon_f^2$$

giving us a way to control the error during the algorithm above.

Furthermore, analogously to the Eckart-Young theorem, [36] proves that, given the bond dimensions D_f and a tensor ψ , there exists a Frobenius-norm-optimal approximation to it in terms of an MPS $\tilde{\psi}$. Analogously to the HOSVD, the truncated TT-SVD does not give the best approximation, but a reasonably good one

$$\|\psi - \psi_{\text{trunc}}\|_F^2 \leq (d-1) \left\| \psi - \tilde{\psi} \right\|_F^2.$$

Arithmetic Addition of two MPS can be achieved by doubling the bond dimensions [36]. Let $A_{i_f}^{(f)}$ and $B_{i_f}^{(f)}$ describe two MPS, where we assume $D_f = D$ for $f = 2, \dots, d$. Then the sum of both MPS is given by

$$C_{i_f}^{(f)} = \begin{pmatrix} A_{i_f}^{(f)} & 0 \\ 0 & B_{i_f}^{(f)} \end{pmatrix}.$$

Oseledets [36] shows that the scalar product of two MPS can be achieved using $O(dnD^3)$ operations; furthermore, matrix-vector products can be computed efficiently if the matrix H is given in a format analogous to MPS

$$H_{i_1 \dots i_d j_1 \dots j_d} = \text{tr} \left[H_{i_1 j_1}^{(1)} \dots H_{i_d j_d}^{(d)} \right],$$

which is a TT representation of an unfolding of H . Suppose the matrices $H_{ij}^{(f)}$ are of size $D' \times D'$. Then the matrix-vector product with a MPS ψ given by $A_{i_f}^{(f)}$ can be obtained from [36]

$$\begin{aligned} (H\psi)_{i_1 \dots i_d} &= \sum_{j_1 \dots j_d} H_{i_1 \dots i_d j_1 \dots j_d} \psi_{j_1 \dots j_d} \\ &\propto \sum_{j_1 \dots j_d} \text{tr} \left[H_{i_1 j_1}^{(1)} \dots H_{i_d j_d}^{(d)} \right] \text{tr} \left[A_{j_1}^{(1)} \dots A_{j_d}^{(d)} \right] \\ &= \text{tr} \left[\left(\sum_{j_1} H_{i_1 j_1}^{(1)} \otimes A_{j_1}^{(1)} \right) \dots \left(\sum_{j_d} H_{i_d j_d}^{(d)} \otimes A_{j_d}^{(d)} \right) \right]. \end{aligned}$$

In other words, the factor matrices $B_{i_f}^{(f)}$ of $H\psi$ are given by tensor products of $H_{i_f j_f}^{(f)}$ and $A_{j_f}^{(f)}$ contracted over j_f . From this, we realize that the bond dimension of $H\psi$ is given by the product of the bond dimensions of H and ψ . If both bond dimensions are given by D , the complexity scales like $O(dn^2 D^4)$. However, usually such a multiplication is followed by a rounding (i. e. reduction of the bond dimension). The idea is to improve the scaling by combining both operations, i. e. not even creating the full matrices $B_{i_f}^{(f)}$ [36].

1.4.6. Tensor Networks

The representations discussed in this section write a “big” tensor in terms of many “small” ones. We can give a graphical representation of this, referred to as an *abstract tensor system* [43] or more commonly *tensor network* [39]. Every tensor is drawn as a circle with one line outwards for every index it has. That is, a vector is a circle with one line, a matrix has two, and so on; see Fig. 1.9. We usually do not label the indices of (i. e. lines leaving a) tensor as in practical use-cases this is obvious.

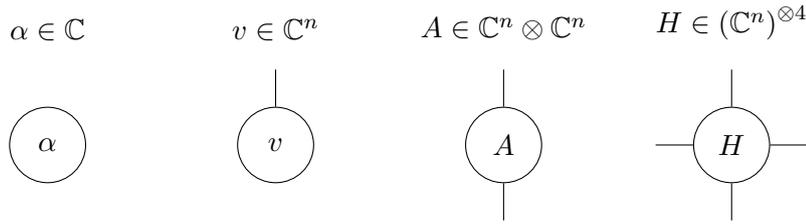


Figure 1.9.: Tensor networks for a scalar α , a vector v , a matrix A , and a tensor H over the vector space \mathbb{C}^n

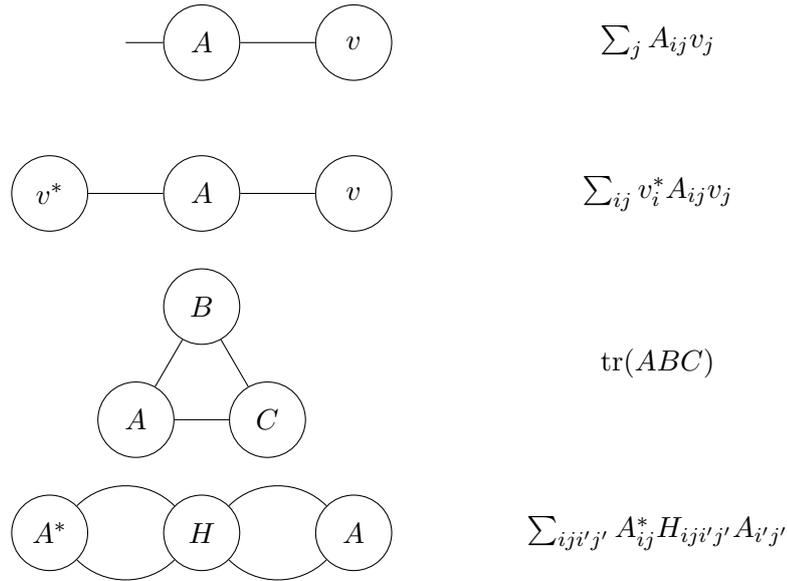


Figure 1.10.: Some examples on arithmetic in tensor networks

This representation enables us to concisely write index contractions as in a matrix-vector multiplication

$$v'_i = \sum_j A_{ij} v_j$$

by connecting the lines representing the corresponding indices of A and v . In other words, an open line in a tensor network represents a free index; a line that connects two tensors represents a summed-over (contracted) index. Fig. 1.10 gives some examples.

With this language established, we are now in the position to write MPS as Tensor Networks, see Fig. 1.11. Fig. 1.12 gives the transfer matrices $E_S^{(f)}$, which become more intuitive in this representation. TI MPS can be represented as tensor networks in an obvious manner.

Strictly speaking, tensor networks may only contain indices that are either free or

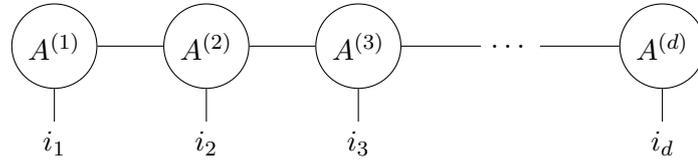


Figure 1.11.: Tensor network representation of a MPS with open boundary conditions

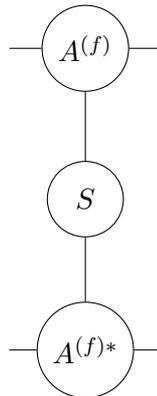


Figure 1.12.: Tensor network for the MPS transfer matrices $E_S^{(f)}$. Indices on the right side of the figure couple to ρ and correspond to the right-hand-side indices of the matrices $A_i^{(f)}$ and $A_i^{(f)*}$. Indices on the left side correspond to $\rho' = E_S^{(f)}(\rho)$.

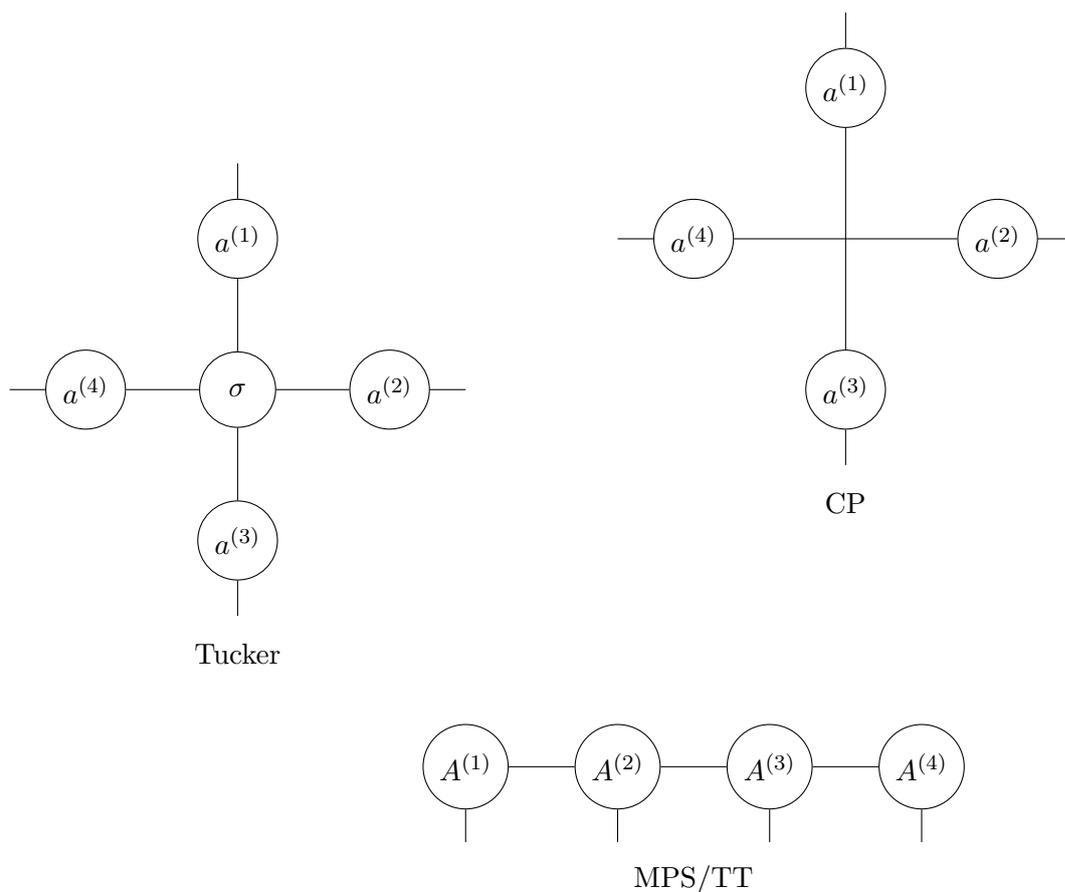


Figure 1.13.: Comparison of MPS/TT, CP and Tucker representations in terms of tensor networks. Note that the canonical decomposition is not strictly a tensor network; however, we can visualize it analogously.

contracted, where contracted means, the index is summed over and appears in exactly two places. In other words, lines do not split up. In this way, we can represent MPS and Tucker forms. However, the third class of tensor decompositions, the CP violates this property. We can nonetheless draw a CP in an analogous way; Fig. 1.13 gives a comparison of the three decompositions for an order-4 tensor.

PEPS Tensor networks give us the ability to naturally generalize MPS to situations beyond a linear ordering of degrees of freedom. One can think of an infinite variety of networks well-suited to specific cases; a particularly important one is the straightforward generalization of MPS to higher dimensions, known as *projected entangled pair states* (PEPS) [39, 44]. These replace the third-order tensors in MPS with fifth-order ones to arrive at a two-dimensional lattice; see Fig. 1.14.

This can be generalized to arrive at PEPS for d sites in S spatial dimensions.

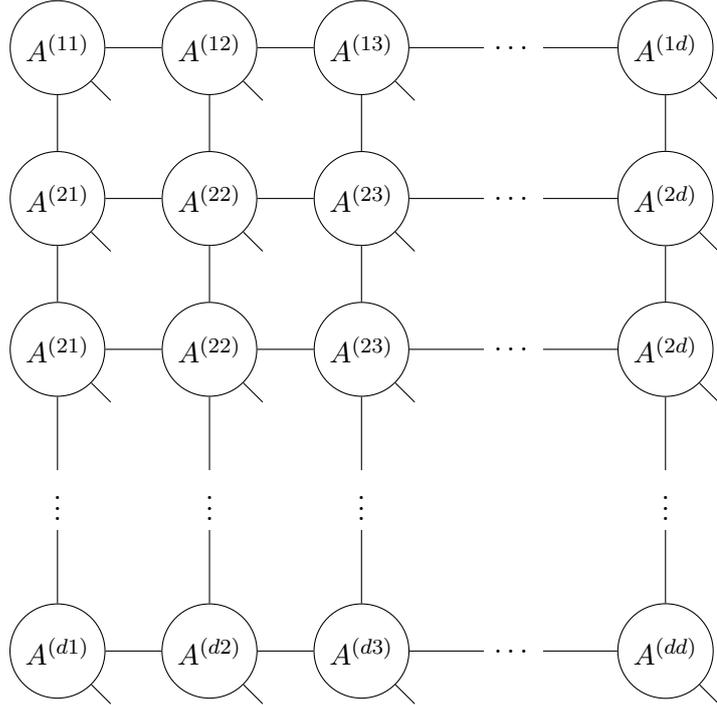


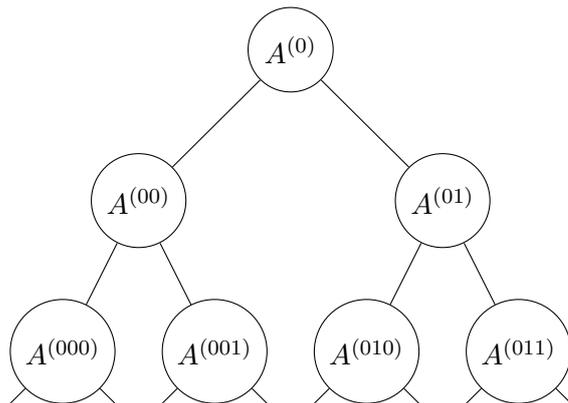
Figure 1.14.: Two-dimensional PEPS state with open boundary conditions

Denoting the bond dimension by D and the one-particle Hilbert space dimension by n , we can deduce the scaling. There are d tensors $A^{(\dots)}$, each of order $1 + 2S$ with shape $n \times D \times \dots \times D$. So, in total we need to store dnD^{2S} complex numbers as opposed to n^d .

Hierarchical Tucker Analogously to \mathcal{H} matrices, we can align the sites of our system on a tree, where the root is the set of all sites and the leaves are sets containing exactly one site. Such a *dimension tree* gives us the tensor in terms of a Hierarchical Tucker representation (\mathcal{H} Tucker) [27, 45]. Again, \mathcal{H} Tucker tensors can be introduced nicely employing tensor networks; see Fig. 1.15.

If we set up a dimension tree where any node has at most one child we see that \mathcal{H} Tucker is a generalization of MPS/TT with open boundary conditions. In particular, any tensor that can be approximated well as a matrix product state also has an efficient \mathcal{H} Tucker representation. Grasedyck [27] presents a truncation algorithm analogous to TT-SVD (i. e. based on the singular value decomposition) and points out that, as opposed to the TT-SVD, this algorithm parallelizes well.

Analogously to PEPS, one can generalize the \mathcal{H} Tucker format to allow for more than one spatial dimension by dropping the requirement that the dimension tree is binary. This idea translates the multidimensional \mathcal{H} matrix approach from [12] to hierarchical Tucker tensors.

Figure 1.15.: \mathcal{H} Tucker format for $d = 8$

Truncation Truncation of an arbitrary tensor to a tensor network following the procedure of TT-SVD, i. e. exploiting the SVD to fix a canonical form, is, in general, no longer possible. This procedure can only be applied if the tensor network in question forms a tree (i. e. it is acyclic) [46], as do MPS or \mathcal{H} Tucker networks; in contrast, PEPS, for example, are not acyclic.

In practice, this issue is of low importance as one would never be able to store a full tensor corresponding to e. g. a PEPS state of practical size. However, the question of canonical forms remains. Evenbly [47] discusses methods to fix the gauge-freedom (choose a canonical form) of an arbitrary tensor network.

1.5. Summary

This part discussed properties, motivation, and applications of three different classes of tensor representation: the canonical decomposition (CP), the Tucker representation, and tensor networks, most notably matrix product states (MPS)/tensor trains (TT). All these representations introduce, in addition to the shape of the tensor, a parameter, the “rank.” Only for the CP, does this “rank” coincide with the tensor rank. The ranks appearing in Tucker generalize row- and column-ranks nicely, whereas ranks in tensor networks are usually referred to as bond dimension.

In all cases a tensor is written as a function of many “smaller” ones, where the ranks determine the sizes of the factor tensors.

Approximation of a tensor is an important tool to gain insights into the structure (geometry) of these approximation. Whereas Tucker and MPS/TT allow truncation in a manner analogous to the Eckart-Young theorem, the problem for CP is in general ill-posed. Not every tensor has a best rank- k approximation; furthermore, the problem, even if it is well-defined, cannot be handled efficiently. Tab. 1.1 gives the headlines of this comparison.

	CP	Tucker	MPS/TT
Rank	Tensor rank r	f -rank r	bond dimension r
Storage	ndr	$ndr + r^d$	ndr^2
Best approximation	ill-posed		exists efficient quasi-best
Application examples	theoretical	Quantum Chemistry	Local interactions with energy gap in 1D

Table 1.1.: Comparison of some properties of the discussed tensor decompositions; we refer to a tensor in $(\mathbb{C}^n)^{\otimes d}$

2. Thoughts on Two Electron Interaction

2.1. Motivation

When propagating Schrödinger atoms or molecules in strong fields, a computationally expensive part is the electron-electron repulsion. All other terms in the Hamiltonian split into tensor products $\text{Id} \otimes H_0 + H_0 \otimes \text{Id}$. This text is concerned with propagating Helium under the approximation of infinite nuclear mass, a six dimensional problem. We work with a discretization (basis of a subspace) of $L^2(\mathbb{R}^6)$ given by

$$\phi_{ilmjpn}(r, \eta, \phi) = f_i(r_1)Y_{lm}(\eta_1, \phi_1)f_j(r_2)Y_{pn}(\eta_2, \phi_2).$$

Y_{lm} denotes the spherical harmonics, $\eta = \cos \theta$ is a more convenient notation for the polar angle. We will often write Ω for both angles. We denote the radial expansion size as $i, j = 1, \dots, N$, and the angular as $l, p = 0, \dots, L - 1$ and $|m| \leq l$, $|n| \leq p$.

After some basic derivations, which will be needed in the following parts, we firstly concern ourselves with the potential in the full basis; afterwards we discuss implications of *constraints* on various methods. By constraints we mean that due to certain symmetries, it is often not necessary to span the full space $l, p < L$ and $|m| \leq l$, $|n| \leq p$. For example, if the Laser pulse is polarized along the z axis, $SO(2)$ symmetry ensures that $m + n$ is conserved.

To evaluate performance of different schemes, we will mostly be concerned with the applications count, denoted as “Appc,” that is, the number of complex multiplications required to apply the potential $\psi' = V\psi$. However, we also need to ensure that all required information fits within our available memory. We will denote the number of complex numbers that need to be stored as “Storage.” To make these numbers more realistic, we assume that a complex number takes 16 bytes of memory (`std::complex<double>` on most machines). Expanding $N = 40$, $L = 20$ yields a 256×10^6 -dimensional discretization; storing a state vector requires about four gigabytes of memory. To gain an intuition as to whether a certain method requires a huge amount of storage, we will compare the amount of memory required in this particular situation.

2.2. Multipole Expansion

We expand the potential $V(\vec{r}_1, \vec{r}_2) = |\vec{r}_1 - \vec{r}_2|^{-1}$ by

$$\begin{aligned}
 V(r_1, \Omega_1, r_2, \Omega_2) &= \sum_{\lambda=0}^{\infty} D_{\lambda}(r_1, r_2) \sum_{\mu=-\lambda}^{\lambda} Y_{\lambda\mu}(\Omega_1) Y_{\lambda\mu}^*(\Omega_2) \\
 &= \sum_{\lambda=0}^{\infty} D_{\lambda}(r_1, r_2) \sum_{\mu=-\lambda}^{\lambda} (-)^{\mu} Y_{\lambda\mu}(\Omega_1) Y_{\lambda-\mu}(\Omega_2),
 \end{aligned}$$

where

$$D_{\lambda}(r_1, r_2) = \frac{4\pi}{2\lambda + 1} \frac{\min(r_1, r_2)^{\lambda}}{\max(r_1, r_2)^{\lambda+1}}.$$

The discretized potential is given by

$$\begin{aligned}
 V_{ilmjpn'l'm'j'p'n'} &= \langle f_{ilm} \otimes f_{jpn} | V | f_{i'l'm'} \otimes f_{j'p'n'} \rangle \\
 &= \sum_{\lambda=0}^{\Lambda-1} [D_{\lambda}]_{ijj'j'} \sum_{\mu=-\lambda}^{\lambda} (-)^{\mu} \langle Y_{lm} | Y_{\lambda\mu} | Y_{l'm'} \rangle \langle Y_{pn} | Y_{\lambda-\mu} | Y_{p'n'} \rangle \\
 &= \sum_{\lambda=0}^{\Lambda-1} [D_{\lambda}]_{ijj'j'} [A_{\lambda}]_{lmpn'l'm'p'n'},
 \end{aligned}$$

defining the matrices A_{λ} . We limit the sum over λ by Λ because, as we will see, certain selection rules imply a maximum on λ .

Zielinski, Majety, and Scrinzi [48] show that for a polynomial basis $f_i(r)$ the matrices D_{λ} ($N^2 \times N^2$ matrices) have maximal rank $2N - 1$ as the polynomials integrated over are of maximal degree $2N - 2$. Transforming to a quadrature grid of size $2N - 1$ gives us D_{λ} in terms of a tensor-product representation and a diagonal matrix. Now it turns out that in practice the quadrature grid may be chosen smaller (N points); therefore, the matrix D_{λ} can be taken diagonal in FEM-DVR [49] (For a detailed derivation of this general fact, see Appendix D in [48]).

The inner products in A_{λ} are subject to selection rules, which we can use to obtain an upper bound on the number of non-zero elements in these matrices. The selection rules for

$$\langle Y_{lm} | Y_{\lambda\mu} | Y_{l'm'} \rangle = (-)^m \sqrt{\frac{(2l+1)(2\lambda+1)(2l'+1)}{4\pi}} \begin{pmatrix} l & \lambda & l' \\ -m & \mu & m' \end{pmatrix} \begin{pmatrix} l & \lambda & l' \\ 0 & 0 & 0 \end{pmatrix},$$

with the Wigner $3j$ symbol, are given by

1. $\mu = m - m'$,
2. $|l - \lambda| \leq l' \leq l + \lambda$,

3. $l + l' + \lambda$ is even.

With that we obtain

$$\begin{aligned} [A_\lambda]_{lm pn l' m' p' n'} &= \sum_{\mu=-\lambda}^{\lambda} \langle Y_{lm} | Y_{\lambda\mu} Y_{l'm'} \rangle \langle Y_{pn} Y_{\lambda\mu} | Y_{p'n'} \rangle \\ &= \langle Y_{lm} | Y_{\lambda(m-m')} Y_{l'm'} \rangle \langle Y_{pn} Y_{\lambda(m-m')} | Y_{p'n'} \rangle, \end{aligned} \quad (2.1)$$

which can be non-zero, if

1. $m - m' = n' - n$,
2. $|m - m'| \leq \lambda$,
3. $l + l' + \lambda$ and $p + p' + \lambda$ are both even,
4. $|l - l'| \leq \lambda \leq l + l'$ and $|p - p'| \leq \lambda \leq p + p'$.

This implies, we require $\Lambda = 2L - 1$ in order to represent the potential exactly.

The scaling of the number of non-zero elements in A_λ is determined solely by the first selection rule, so we obtain (for expansion $l, p, l', p' = 0, \dots, L - 1$, i.e. the matrix size will scale like $L^4 \times L^4$) that the number of non-zero elements in A_λ scales like L^7 or matrix dimension to the 7/4 with respect to the angular basis. Since the selection rule does not depend on λ , the scaling of the number of non-zeros in V is also given by L^7 .

2.3. Unconstrained Basis

In this section we assume a full discretization of the angular part: $l, p < L$, $|m| \leq l$, $|n| \leq p$. The radial part is of dimension N^2 giving us a $L^4 N^2$ -dimensional discretization.

2.3.1. Straight-Forward Scheme

As we have seen in the previous section, we can apply V by storing the matrices D_λ and A_λ for $\lambda = 0, \dots, \Lambda - 1$. We can then apply V (without exploiting any structure within A_λ) with applications count

$$\text{Appc}_{F1}(L, N)/N^2 = \sum_{\lambda=0}^{\Lambda-1} |A_\lambda|,$$

where we denote by $|A_\lambda|$ the number of non-zero elements in the matrix A_λ . We will denote this application scheme with the index $F1$ (F standing for full). This applications count is actually too low as in reality for every application of D_λ this

matrix needs to be multiplied by the corresponding $[A_\lambda]_{lpmn'l'm'n'}$. However, we will not take this into account as in the end, we are mostly interested in the scaling of the methods.¹

Let us investigate the storage requirements in this scheme. This is determined by two parts: The diagonal matrices D_λ and the full matrices A_λ . Notice that, with negligible overhead, we can calculate the matrix elements of A_λ by only storing the Gaunt coefficients $\langle Y_{lm}|Y_{\lambda\mu}|Y_{l'm'}\rangle$. These can be obtained by quadrature from P_l^m , the associated Legendre functions, stored at the quadrature points. Assuming a $\max(\Lambda, L)$ point quadrature rule, we need to store $\max(\Lambda, L)^3/2$ complex numbers (actually twice as many real numbers, but for the sake of consistency, we give storage in terms of complex numbers). In total, the potential requires in this scheme

$$\text{Storage}_{F1}(L, N) = \Lambda N^2 + \frac{1}{2}(\max(\Lambda, L))^3$$

complex numbers. Fixing $\Lambda = 2L - 1$, we obtain roughly $2LN^2 + 4L^3$ complex numbers. This enables us, at $N = 40$, $L = 20$, to store the potential in less than two megabytes of RAM. If we were to store all Gaunt coefficients instead of the associated Legendre functions (replacing the power of three by five, where we make use of the conservation of magnetic momentum), the same problem size would need about 800 megabytes to be stored, which is still an unproblematic size.

2.3.2. Explicit Scheme

Let us compare this to a scheme, where we store the matrix explicitly only taking advantage of the selection rule $m - m' = n' - n$ and the diagonality in the radial indices. We will denote this with the index $F2$. The applications count $\text{Appc}_{F2}(L, N)/N^2$ is given by the number of index pairs $lmpnl'm'p'n'$ for which at least one A_λ has a non-zero entry. Storage requires

$$\text{Storage}_{F2}(L, N, \Lambda) = L^7 N^2.$$

The previously mentioned example situation $N = 40$, $L = 20$ can be stored in RAM if we have at least 32 terabytes at our disposal. This is not the case for the machines used in writing this text.

Let us nonetheless compare these two application schemes, where we count the number of non-zero elements numerically. The results are given in Fig. 2.1. As expected, the scheme $F2$ performs better, scaling like L^7 . The scheme $F1$ appears to be scaling like L^8 in leading order, which suggests, there are no other constraints enhancing the scaling than conservation of magnetic momentum. However, the matrices are quite sparse.

¹Additionally, these are not complex multiplications as the Wigner $3j$'s as well as the matrices D_λ are real.

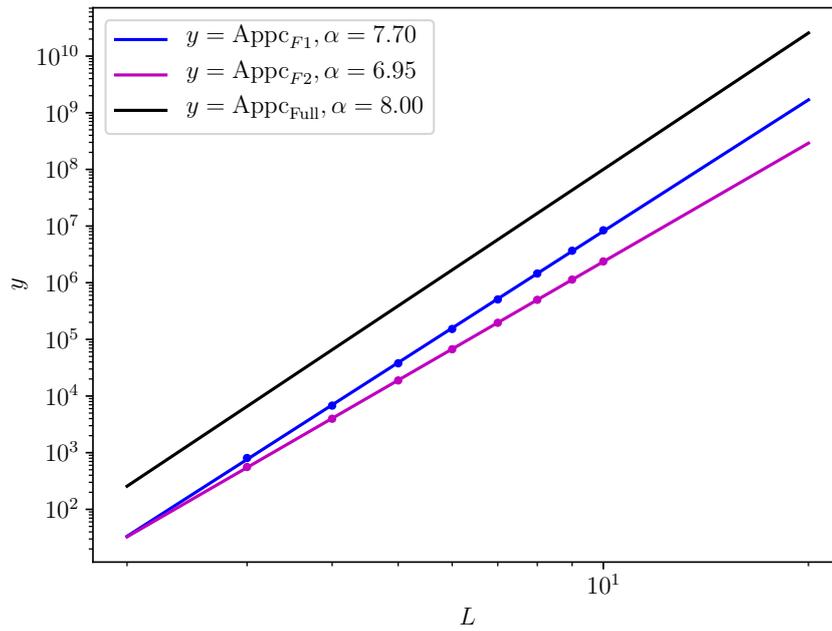


Figure 2.1.: Comparison of the schemes $F1$ and $F2$; “Full” is given by L^8 . Fits are defined by $\log y = \alpha \log L + \beta$ taking into account all plotted data points, orders α are given.

2.3.3. Tensor Product Scheme

Equation (2.1) looks suspiciously like a tensor product. In fact, we can write the application of A_λ as

$$\psi'_{lmpn} = \left[B_\lambda^{(1)} \right]_{lml'm'} \left[B_\lambda^{(2)} \right]_{pnp'n'} \delta_{(m-m')(n'-n)} \psi_{l'm'p'n'}.$$

This is not strictly a tensor product. To split the application in two steps, as we do for tensor products, we need to retain memory of $m - m'$ resp. $n - n'$. This is achieved in

$$\begin{aligned} \phi_{l'm'pn\mu} &= \sum_{p'n'\mu} \delta_{\mu(n'-n)} \left[B_\lambda^{(2)} \right]_{pnp'n'} \psi_{l'm'p'n'}, \\ \psi'_{lmpn} &= \sum_{l'm'\mu} \delta_{\mu(m-m')} \left[B_\lambda^{(1)} \right]_{lml'm'} \phi_{l'm'pn\mu}. \end{aligned}$$

Both steps can be performed without explicitly summing over μ as the value is determined by n and n' resp. m and m' . Therefore, the apply count for either step is $L^2 \left| B_\lambda^{(1/2)} \right|$. With $\left| B_\lambda^{(1)} \right| = \left| B_\lambda^{(2)} \right|$, we have

$$\text{Appc}_P(L, N)/N^2 = 2L^2 \sum_{\lambda=0}^{\Lambda-1} \left| B_\lambda^{(1)} \right|,$$

where P stands for “product.”

Inserting the expression

$$\left[B_\lambda^{(1)} \right]_{lml'm'} = \langle Y_{lm}, Y_{\lambda(m-m')} Y_{l'm'} \rangle,$$

we can deduce the scaling as $\left| B_\lambda^{(1)} \right| \in O(L^4)$ as the selection rule on the magnetic momenta is always satisfied. Therefore, for the full multipole expansion $\Lambda = 2L - 1$ we expect $\text{Appc}_P(L, N)/N^2$ to scale like $O(L^7)$. Fig. 2.2 compares the applications count to the schemes $F1$ and $F2$. The product scheme scales slightly better than the full schemes, beating $F1$ at $L \approx 10$.

Storage-wise the scheme P does not differ from $F1$ as we need the Gaunt coefficients as well as the radial matrices D_λ . Therefore, the example situation can be stored in a negligible amount of memory.

2.3.4. Product Grid

As the Coulomb potential is a multiplication operator, it is diagonal in a grid basis. A transformation to a grid in η and ϕ can be achieved by a tensor product. So, it seems like a viable alternative to transform to a six-dimensional grid, apply the potential there, and transform back. We assume, as is the case in our implementation, that the radial basis is already a grid and requires no transformation. Let us investigate the

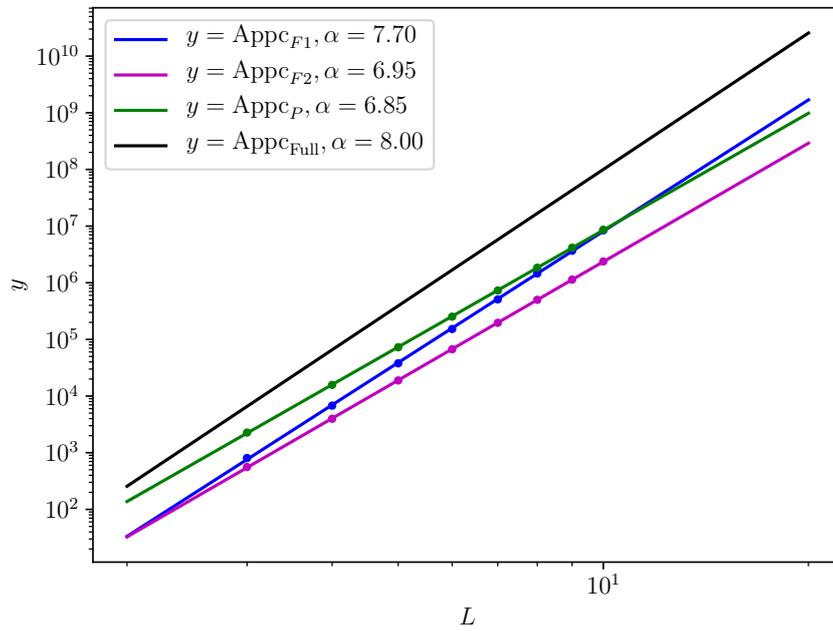


Figure 2.2.: Apply count for the scheme P compared to the full schemes $F1$ and $F2$. “Full” is given by L^8 .

transformation from the spherical harmonics basis to a (product) quadrature grid. For a large enough grid we will obtain a diagonal transformed matrix. Denoting the new grid indices with $\alpha, \beta, \gamma, \delta$ and the grid basis functions with $P_\alpha^{\eta/\phi}$ (corresponding to quadrature points $q_\alpha^{\eta/\phi}$ with weights $w_\alpha^{\eta/\phi}$), where we assume the same grid for both electrons, we get the transformation

$$\begin{aligned} T_{\alpha\beta\gamma\delta l m p n} &= \langle P_\alpha^\eta | P_l^m \rangle \langle P_\beta^\phi | e^{im\phi} \rangle \langle P_\gamma^\eta | P_{pn} \rangle \langle P_\delta^\phi | e^{in\phi} \rangle \\ &= T_{\alpha\beta l m}^{(1)} T_{\gamma\delta p n}^{(1)}, \end{aligned}$$

with the associated Legendre functions P_l^m . The transformed potential will be given as

$$V_{\alpha\beta\gamma\delta\alpha'\beta'\gamma'\delta'} = \delta_{\alpha\alpha'} \delta_{\beta\beta'} \delta_{\gamma\gamma'} \delta_{\delta\delta'} V_{\alpha\beta\gamma\delta}.$$

Let us denote the grid sizes by G_l, G_m , where we have the minimal requirement $G_l \geq L, G_m \geq 2L - 1$ to obtain a unitary transformation. Transforming to a quadrature grid and back yields the same result as calculating all matrix elements using a quadrature of order G_l, G_m . As the multipole expansion is truncated by the selection rules at $\Lambda = 2L - 1$, the η integrals are over a polynomial of degree at most $2(L - 1) + (\Lambda - 1) = 4(L - 1)$. Gaussian quadrature is exact for polynomials of degree up to $2G_l - 1$ and we arrive at the condition $G_l \geq 2L - 1$ for exact matrix elements. (However, as pointed out in [48] for the radial grid, where we make use of the same trick, this enlarging of the quadrature might not be necessary in practice.)

The quadrature in ϕ direction is given by equidistant points with weights $1/G_m$ (DFT). Notice that we reproduce the integral, for $m \neq 0$,

$$0 = \int_0^{2\pi} \exp[im\phi] d\phi = \frac{1}{G_m} \sum_{k=0}^{G_m-1} \exp\left[im \frac{2\pi k}{G_m}\right] = \frac{1}{G_m} \frac{1 - \exp[im2\pi]}{1 - \exp\left[im \frac{2\pi}{G_m}\right]}$$

if $m/G_m \notin \mathbb{Z}$. The largest value of $|m|$ in this integral is given by $(L - 1) + (L - 1) + (2L - 2) = 4(L - 1)$; so, we assume to need a quadrature grid about twice as large as the ϕ discretization as well.

The apply count of a diagonal V is then given by the matrix dimension $G_l G_m G_p G_n$, which scales like L^4 . Let us now investigate the applications count of the transformation for one electron

$$\begin{aligned}
 y_{\alpha\beta} &= \sum_{l=0}^{L-1} \sum_{m=-l}^l T_{\alpha\beta lm}^{(1)} y_{lm} \\
 &= \sum_{l=0}^{L-1} \sum_{m=-l}^l T_{\alpha l}^{(\eta,m)} T_{\beta m}^{(\phi)} y_{lm} \\
 &= \sum_{m=-L}^{L-1} T_{\beta m}^{(\phi)} \sum_{l=|m|}^{L-1} T_{\alpha l}^{(\eta,m)} y_{lm}.
 \end{aligned}$$

The inner sum (with indices αm) consists of G_l elements for each $m = -L+1, \dots, L-1$. One such element can be calculated with operations count $L - |m|$. So, in total, we have operations count $G_l L^2$ to perform the inner sum. The outer sum can be performed with operations count $2L - 1$ for each of the $G_l G_m$ elements. That is, in total the one electron transformation can be done with operations count

$$\text{Appc} \left[T^{(1)} \right] = G_l L^2 + (2L - 1) G_l G_m,$$

which gives the expected tensor product L^3 scaling. The two electron transformation is a tensor product of two of these transformations. The apply count of a tensor product of two $M \times N$ matrices A is given by $\text{Appc} [A \otimes A] = (M + N) \text{Appc} [A]$. Therefore, we obtain the operations count

$$\begin{aligned}
 \text{Appc} [T] &= (G_l G_m + L^2) (G_l L^2 + (2L - 1) G_l G_m) \\
 &= G_l^2 G_m L^2 + G_l^2 G_m^2 (2L - 1) + G_l L^4 + G_l G_m L^2 (2L - 1).
 \end{aligned}$$

Let us introduce $\kappa_l = G_l/L$ and $\kappa_m = G_m/(2L - 1)$. We say *full grid* meaning $\kappa_l = \kappa_m = 2$ and *minimal grid* meaning $\kappa_l = \kappa_m = 1$. With that, the apply count reads

$$\text{Appc} [T] = \kappa_l L^5 + \kappa_l^2 \kappa_m L^4 (2L - 1) + \kappa_l \kappa_m L^3 (2L - 1)^2 + \kappa_l^2 \kappa_m^2 L^2 (2L - 1)^3,$$

recovering the expected L^5 scaling.

Denoting this application scheme as $G1$ (G standing for grid), we arrive at

$$\begin{aligned}
 \text{Appc}_{G1}(L, N)/N^2 &= 2\kappa_l L^5 + 2\kappa_l^2 \kappa_m L^4 (2L - 1) + 2\kappa_l \kappa_m L^3 (2L - 1)^2 \\
 &\quad + 2\kappa_l^2 \kappa_m^2 L^2 (2L - 1)^3 + \kappa_l^2 \kappa_m^2 L^2 (2L - 1)^2.
 \end{aligned}$$

The leading order L^5 has the prefactor

$$(2\kappa_l + 4\kappa_l^2 \kappa_m + 8\kappa_l \kappa_m + 16\kappa_l^2 \kappa_m^2) \in [30, 324]$$

for any grid between a minimal and a full grid. This means that, although the scaling is significantly improved when transforming to a quadrature grid, the enormous prefactor might render this scheme undesirable for realistic matrix dimensions. The situation is plotted in Fig. 2.3.

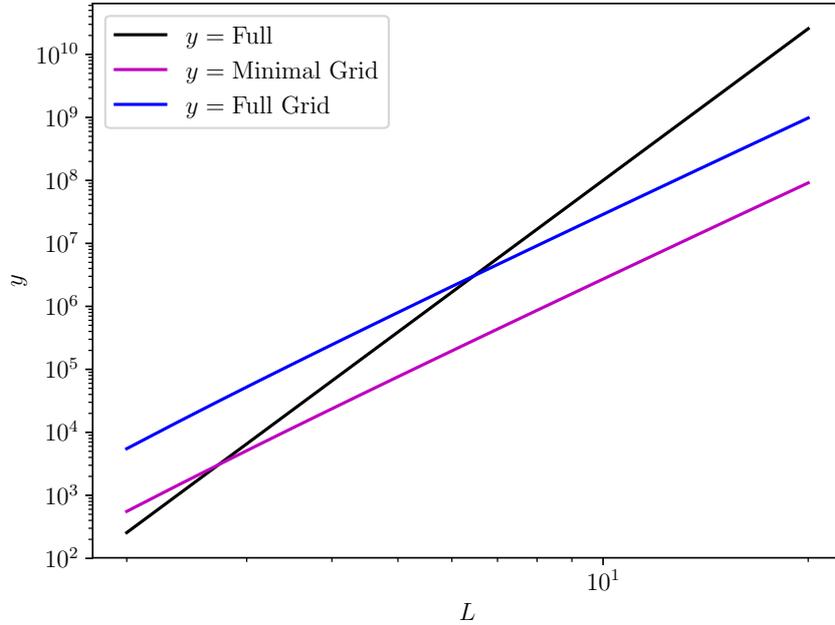


Figure 2.3.: Apply count for the scheme G_1 . “Full” is given by L^8 .

The storage requirements of this scheme are given by L^2 associated Legendre functions on G_l grid points as well as $(2L - 1)$ ϕ basis functions on G_m grid points and the diagonal potential at $G_l^2 G_m^2 N^2$ points. We have

$$\text{Storage}_{G_1}(L, N) = \kappa_l L^3 + \kappa_m (2L - 1)^3 + \kappa_l^2 \kappa_m^2 L^2 (2L - 1)^2 N^2,$$

and we can fit the potential in the example $L = 20$, $N = 40$, and $\kappa_m = \kappa_l = 2$ into 250 gigabytes of memory.

2.3.5. η Grid

One thing that is not satisfying about the grid transformation is the fact that the conservation rule $m - m' = n' - n$ is not employed explicitly. Instead, we transform to a full grid and back recovering a matrix that satisfies this rule. Starting from this point of view, let us investigate a grid transformation where we only transform the coordinate η ; this way, we should still be able to make use of the conservation rule. Recall the full potential assumed to be diagonal in the radial indices

$$\begin{aligned}
 & V_{ilmjpn'i'l'm'j'p'n'} \\
 &= \delta_{ii'}\delta_{jj'} \sum_{\lambda=0}^{\Lambda-1} \sum_{\mu=-\lambda}^{\lambda} (-)^{\mu} [D_{\lambda}]_{ij} \langle Y_{lm} | Y_{\lambda\mu} | Y_{l'm'} \rangle \langle Y_{pn} | Y_{\lambda-\mu} | Y_{p'n'} \rangle \\
 &= \delta_{ii'}\delta_{jj'} \sum_{\lambda=0}^{\Lambda-1} \sum_{\mu=-\lambda}^{\lambda} (-)^{\mu} [D_{\lambda}]_{ij} \delta_{\mu(m-m')} \delta_{\mu(n'-n)} \langle P_l^m | P_{\lambda}^{\mu} | P_{l'}^{m'} \rangle \langle P_p^n | P_{\lambda}^{-\mu} | P_{p'}^{n'} \rangle \\
 &= \delta_{ii'}\delta_{jj'} \sum_{\mu=-\Lambda+1}^{\Lambda-1} \delta_{\mu(m-m')} \delta_{\mu(n'-n)} (-)^{\mu} \sum_{\lambda=-|\mu|}^{\Lambda-1} [D_{\lambda}]_{ij} \langle P_l^m | P_{\lambda}^{\mu} | P_{l'}^{m'} \rangle \langle P_p^n | P_{\lambda}^{-\mu} | P_{p'}^{n'} \rangle,
 \end{aligned}$$

with appropriately normalized P_l^m . Suppose we can apply the matrices $\langle P_l^m | P_{\lambda}^{\mu} | P_{l'}^{m'} \rangle$ by transforming to a grid of size κL

$$\langle P_l^m | P_{\lambda}^{\mu} | P_{l'}^{m'} \rangle \delta_{\mu(m-m')} = \left(T_m^{\dagger} \right)_{l\alpha} [P_{\lambda}^{\mu}]_{\alpha} (T_{m'})_{\alpha l'} \delta_{\mu(m-m')},$$

where $T_m \in \mathbb{C}^{\kappa L \times (L-|m|)}$. Then

$$\begin{aligned}
 & V_{ilmjpn'i'l'm'j'p'n'} \\
 &= \delta_{ii'}\delta_{jj'} \sum_{\mu=-\Lambda+1}^{\Lambda-1} \delta_{\mu(m-m')} \delta_{\mu(n'-n)} (-)^{\mu} \\
 & \quad \sum_{\lambda=-|\mu|}^{\Lambda-1} [D_{\lambda}]_{ij} \left(T_m^{\dagger} \right)_{l\alpha} [P_{\lambda}^{\mu}]_{\alpha} (T_{m'})_{\alpha l'} \left(T_n^{\dagger} \right)_{p\beta} [P_{\lambda}^{-\mu}]_{\beta} (T_{n'})_{\beta p'} \\
 &= \delta_{ii'}\delta_{jj'} \sum_{\mu=-\Lambda+1}^{\Lambda-1} \delta_{\mu(m-m')} \delta_{\mu(n'-n)} (-)^{\mu} \left(T_m^{\dagger} \right)_{l\alpha} \left(T_n^{\dagger} \right)_{p\beta} \\
 & \quad \sum_{\lambda=-|\mu|}^{\Lambda-1} [D_{\lambda}]_{ij} [P_{\lambda}^{\mu}]_{\alpha} [P_{\lambda}^{-\mu}]_{\beta} (T_{m'})_{\alpha l'} (T_{n'})_{\beta p'} \\
 &= \delta_{ii'}\delta_{jj'} \sum_{\mu=-\Lambda+1}^{\Lambda-1} \delta_{\mu(m-m')} \delta_{\mu(n'-n)} \left(T_m^{\dagger} \right)_{l\alpha} \left(T_n^{\dagger} \right)_{p\beta} [K_{\mu}]_{ij\alpha\beta} (T_{m'})_{\alpha l'} (T_{n'})_{\beta p'} \\
 &= \delta_{ii'}\delta_{jj'} \delta_{(m-m')(n'-n)} \left(T_m^{\dagger} \right)_{l\alpha} \left(T_n^{\dagger} \right)_{p\beta} [K_{(m-m')}]_{ij\alpha\beta} (T_{m'})_{\alpha l'} (T_{n'})_{\beta p'},
 \end{aligned}$$

where

$$[K_\mu]_{ij\alpha\beta} = (-)^\mu \sum_{\lambda=|\mu|}^{\Lambda-1} [D_\lambda]_{ij} [P_\lambda^\mu]_\alpha [P_\lambda^{-\mu}]_\beta.$$

In this scheme, we can apply the operator $\phi_{ilmjpn} = V_{lm\mu n l' m' p' n'}^{ij} \phi_{il' m' j p' n'}$ following the procedure:

1. Transform

$$\psi_{i\alpha m' j \beta n'} = \sum_{l' p'} (T_{m'})_{\alpha l'} (T_{n'})_{\beta p'} \phi_{il' m' j p' n'}.$$

To achieve this, we need to apply $N^2(2L-1)^2$ tensor products $T_{m'} \otimes T_{n'}$.

2. Set $\psi'_{i\alpha m j \beta n} \leftarrow 0$.

3. For every m, m', n :

- a) Set $n' = m + n - m'$; if $n' \notin \{-L+1, \dots, L-1\}$, skip the next step.
- b) Apply the diagonal matrix $K_{(m-m')}$ on the subvector of ψ of fixed m, n, m', n' adding the results into

$$\psi'_{i\alpha m j \beta n} \leftarrow \psi'_{i\alpha m j \beta n} + [K_{(m-m')}]_{ij\alpha\beta} \psi_{i\alpha m' j \beta n'}.$$

Notice the absence of sums in this step; the iteration over all magnetic momenta constitutes a sum, however.

This step requires the application of $O(L^3)$ diagonal matrices of dimension $N^2(\kappa L)^2$.

4. Transform back to spherical harmonics

$$\phi_{ilmjpn} = \sum_{\alpha\beta} (T_m^\dagger)_{l\alpha} (T_n^\dagger)_{p\beta} \psi'_{i\alpha m j \beta n}.$$

To investigate the applications count in this scheme, we note that a tensor product $A \otimes B$, with $A \in \mathbb{C}^{m \times m'}$, $B \in \mathbb{C}^{n \times n'}$ can be applied using

$$\text{Appc}[A \otimes B] = m'mn + m'nn'.$$

Hence, the first step and last step each take

$$\begin{aligned} \text{Appc}_1(L, N)/N^2 &= \sum_{m'=-L+1}^{L+1} \sum_{n'=-L+1}^{L-1} \text{Appc}[T_{m'} \otimes T_{n'}] \\ &= \sum_{m'=-L+1}^{L+1} \sum_{n'=-L+1}^{L-1} (\kappa^2 L^2 (L - |m'|) + \kappa L (L - |m'|)(L - |n'|)). \end{aligned}$$

The application of the diagonal part takes

$$\text{Appc}_2(L, N)/N^2 = \sum_{m=-L+1}^{L-1} \sum_{n=-L+1}^{L-1} \sum_{m'=-L+1}^{L-1} \chi_{[-L+1, L-1]}(m+n-m') \kappa^2 L^2.$$

We have

$$\text{Appc}_{G_2}(L, N) = 2\text{Appc}_1(L, N) + \text{Appc}_2(L, N).$$

Instead of doing the calculation explicitly, we numerically compute a few values and fit a polynomial to obtain the prefactors. We get L^5 scaling as expected, with a prefactor in $[34/3, 124/3]$ for a grid between $\kappa = 1$ and $\kappa = 2$.

Let us evaluate the storage requirements of this scheme. We need $(2L-1)$ transformations of size $\kappa L \times (L-|m|)$; also, we need to store K_μ for any value of $\mu = m - m'$, that is, $4L - 3$ different vectors of size $\kappa^2 L^2 N^2$. In total

$$\text{Storage}_{G_2}(L, N) = \kappa L^3 + (4L - 3)\kappa^2 L^2 N^2.$$

Again, for $\kappa = 2$, $L = 20$, and $N = 40$ this scheme requires about three gigabytes of RAM, which certainly seems feasible.

Fig. 2.4 compares apply counts of both a full and a minimal grid in η with the full matrix L^8 .

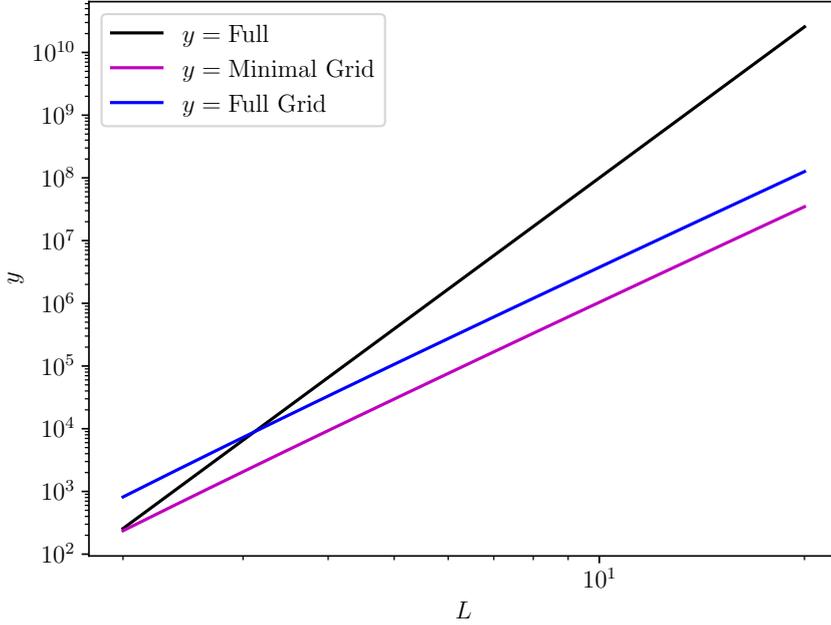


Figure 2.4.: Apply count for the scheme G_2 . “Full” is given by L^8 .

2.3.6. Thoughts on Grid Sizes

As we have seen, the grid transformation $G2$ beats the product grid $G1$ in terms of apply counts; so we will not investigate ϕ grid sizes any further. However, the question remains, can we represent $\langle P_l^m | P_\lambda^\mu | P_{l'}^{m'} \rangle$ on a sufficiently small quadrature grid. Firstly, we note that P_l^m are not polynomials; however, looking at the definition of the associated Legendre functions

$$P_l^m(\eta) \propto (1 - \eta^2)^{|m|/2} \frac{d^{|m|}}{d\eta^{|m|}} P_l(\eta),$$

we see that one such integral $\langle P_l^m | P_\lambda^\mu | P_{l'}^{m'} \rangle$ contains a polynomial of degree $(l - |m|) + (\lambda - |\mu|) + (l' - |m'|)$ times $(1 - \eta^2)$ to the power of $(|m| + |m'| + |\mu|)/2$. Note that $\mu = m - m'$ implies that $|m| + |m'| + |\mu|$ is even; that is, all non-zero contributions in the expansion are integrals over a polynomial of degree $l + l' + \lambda$, justifying the use of Gauß-Legendre quadrature. In some sense, the selection rule “protects” Gauß-Legendre quadrature from unfit integrands.

To test these considerations in practice, we calculate the matrix elements $\langle P_l^m | P_\lambda^\mu | P_{l'}^{m'} \rangle$ with both κL and $2\kappa L$ quadrature and compare the results v_1 and v_2 . We plot the error $\epsilon = |v_1 - v_2| / (|v_1| + |v_2| + 1)$. Fig. 2.5 shows the errors for all matrix elements; that is, we do not make use of the protection $|m| + |m'| + |\mu| = 2n$. We see that the quadrature does not converge in any satisfactory way and we indeed get only correct matrix elements if the selection rule is satisfied.

Fig. 2.6 shows the quadrature errors of the matrix elements, which satisfy all Wigner-3j selection rules. We see that $\kappa \approx 2$ is necessary and probably going to a smaller quadrature grid yields substantial errors. With this in mind, we will always set $\kappa = 2$ for the method G_2 , and $\kappa_l = \kappa_m = 2$ for $G1$.

An open question remaining is, can we choose another quadrature than Gauß-Legendre to represent the matrix elements on a smaller grid. For example, one might pick a quadrature corresponding to the weight function $(1 - \eta^2)^n$ for some n . With choosing such a quadrature, however, one ends up integrating functions that are not polynomials (e. g. $(1 - \eta^2)^{-n}$ for $l = l' = \lambda = 0$); the problem with this is that we need *one* quadrature grid for *all* values of λ . This suggests that any other quadrature than Gauß-Legendre will not be able to represent all matrix elements exactly.

2.3.7. Coupled Angular Momenta

From $SO(3)$ representation theory we know that the space $A(l) \otimes A(p)$, where we denote $A(l) = \text{span}(\{Y_{lm} : -l \leq m \leq l\})$, can be decomposed as a direct sum

$$A(l) \otimes A(p) = \bigoplus_{J=|l-p|}^{l+p} A(J),$$

which is interpreted physically as coupling of angular momenta. In our case, we can use this to transform our space spanned by the angular basis functions

$$\left(\bigoplus_{l=0}^{L-1} A(l) \right) \otimes \left(\bigoplus_{p=0}^{L-1} A(p) \right) = \bigoplus_{l,p=0}^{L-1} \left(\bigoplus_{J=|l-p|}^{l+p} A(J) \right).$$

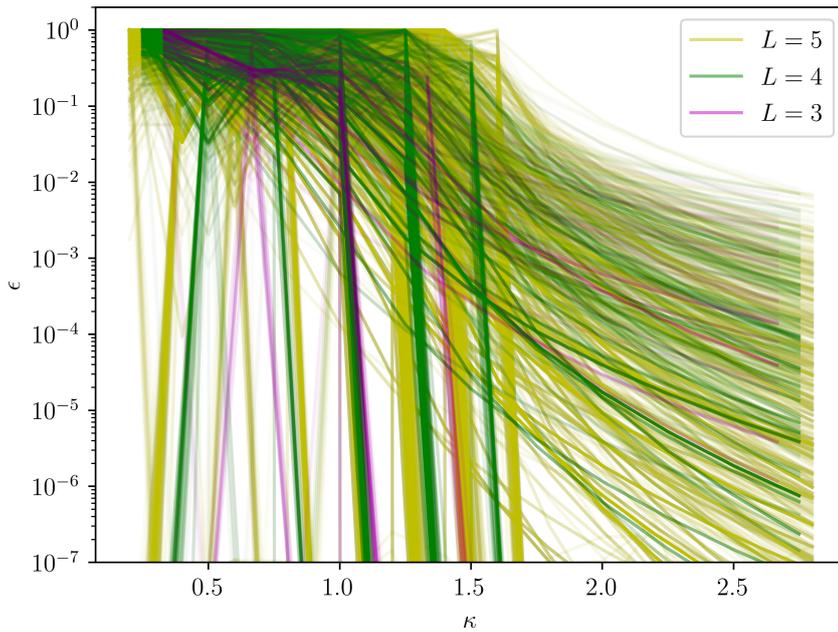


Figure 2.5.: Quadrature errors ϵ of the associated Legendre matrix elements; each dataset corresponds to one matrix element evaluated with different quadratures. Selection rules are not enforced, meaning the integration does not happen over polynomials, explaining the substantial errors remaining even for high order quadratures.

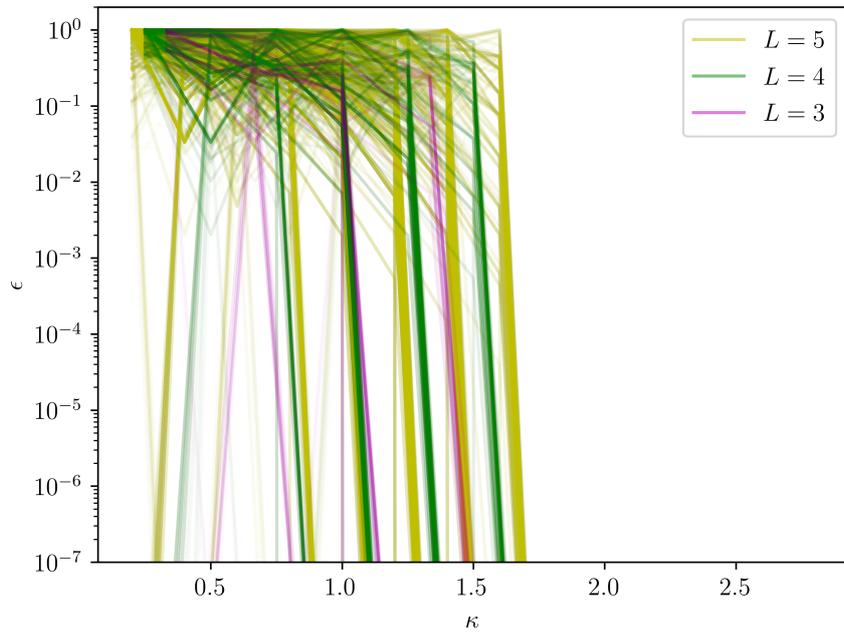


Figure 2.6.: Quadrature errors ϵ of the associated Legendre matrix elements; each dataset corresponds to one matrix element satisfying the selection rules evaluated with different quadratures.

The reasoning behind this is that the $SO(3)$ symmetry of the Coulomb potential implies, the transformed matrix will be diagonal in the coupled angular momentum J and z component M . However, diagonality in lp will not be achieved, leaving us with a L^6 or matrix dimension to the 6/4 scaling. The transformation is given by the Clebsch-Gordan coefficients written as

$$\langle lm, pn | lp, JM \rangle = (-)^{l-p+M} \sqrt{2J+1} \begin{pmatrix} l & p & J \\ m & n & -M \end{pmatrix}.$$

With that, we obtain for the transformed matrices A_λ

$$[A_\lambda]_{lpJMl'p'J'M'} = \langle lp, JM | lm, pn \rangle \langle Y_{lm} | Y_{\lambda(m-m')} Y_{l'm'} \rangle \langle Y_{pn} Y_{\lambda(m-m')} | Y_{p'n'} \rangle \langle l'm', p'n' | l'p', J'M' \rangle,$$

summations over m, n, m', n' implied. The selection rules then imply $m-m' = n-n'$ and $m+n = M, m'+n' = M'$ or $M = M'$ and $m+n = m'+n' = M$ collapsing the sums over n and n' :

$$[A_\lambda]_{lp'l'p'}^{JM} = \langle lp, JM | lm, p [M-m] \rangle \langle Y_{lm} | Y_{\lambda(m-m')} Y_{l'm'} \rangle \langle Y_{p[M-m]} Y_{\lambda(m-m')} | Y_{p'[M-m']} \rangle \langle l'm', p' [M-m'] | l'p', JM \rangle,$$

with remaining summations over m and m' .

Let us now investigate, how the transformation from the tensor product to the coupled basis scales with L

$$T_{lmpn'l'p'J'M'} = \delta_{ll'} \delta_{pp'} \langle lm, pn | lp, J'M' \rangle,$$

subject to the Wigner $3j$ selection rules. For fixed $lmpn$, only $M = m+n$ gives non-zero entries. Furthermore, the matrix is diagonal in two indices leaving us with L^5 , or matrix dimension to the 5/4, scaling.

In summary, these results suggest that we can move from 7/4 scaling to 6/4 scaling if we transform with cost scaling like 5/4. The transformation is diagonal in the one-particle angular momenta lp , and the transformed potential is diagonal in the coupled angular momenta JM .

To gain more detailed insights, we define

$$\begin{aligned} \text{Appc}_T(L, N) &= |T|, \\ \text{Appc}_{C_1}(L, N) &= \sum_{\lambda=0}^{\Lambda-1} \left| T^\dagger A_\lambda T \right|, \\ \text{Appc}_{C_2}(L, N) &= \left| T^\dagger V T \right|, \end{aligned}$$

defining the schemes $C1$ and $C2$ analogously to the schemes $F1$ and $F2$ (C standing for coupled). We exclude the transformation from the apply count of both schemes, as a realistic implementation of either scheme would set up the whole Hamiltonian in the coupled basis. Results are plotted in Fig. 2.7. Again, switching from the multipole scheme $C1$ to $C2$ costs almost an order in scaling; also, analogous to $F1$ and $F2$ the matrices are quite sparse.

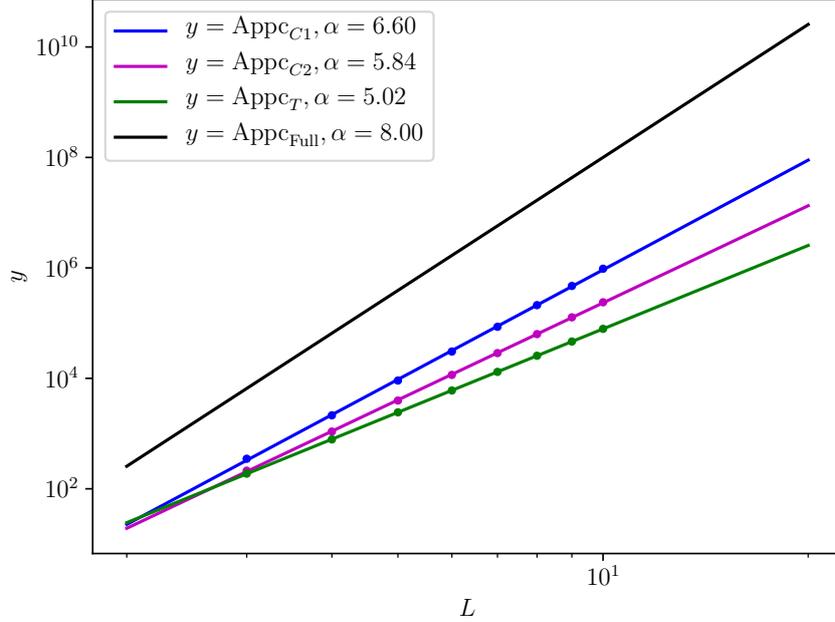


Figure 2.7.: Applications count of the schemes $C1$ and $C2$. “Full” is given by L^8 ; fits are affine in log-scale with orders given.

In the scheme $C1$ we need to store the matrices $[A_\lambda]^{JM}$ for $J = 0, \dots, 2L - 2$ as well as the diagonals D_λ for $\lambda = 0, \dots, \Lambda - 1$. Therefore, we have

$$\text{Storage}_{C1}(L, N) = \Lambda N^2 + \Lambda(2L - 1)^2 L^4.$$

The scheme $C2$ requires us to store the full matrix, where we make use of the fact that it is diagonal in the radial indices and JM . Hence

$$\text{Storage}_{C2}(L, N) = N^2(2L - 1)^2 L^4,$$

or, for our example situation, about 150 gigabytes for the scheme $C1$ and roughly six terabytes for $C2$. Although the storage requirements of $C1$ are not as nice as for the scheme $F1$, it lies still within the range of feasibility. $C2$, however, is currently not possible with the machines at our disposal.

One has to remark that probably a better method can be found to store the transformed matrices A_λ in the scheme C1. Carrying out the sums over m and m' explicitly might even be a performance penalty one is willing to pay as the resulting value $[A_\lambda]_{lp'p'}^{JM}$ multiplies the diagonal matrix D_λ . Therefore, any penalty in calculating that value is suppressed by $1/N^2$ compared to the actual application.

2.3.8. Further Ideas

In the spirit of the idea “we do not need to know the interaction in great detail at places in configuration space, where the interaction is weak and simple,” we reason that one can truncate the multipole expansion at different Λ , depending on the radial patch ij . In other words, where both electrons are well-separated, the interaction should be sufficiently well approximated by small Λ . To investigate this more precisely, we recall the radial functions in the multipole expansion

$$D_\lambda(r_1, r_2) = \frac{4\pi}{2\lambda + 1} \frac{\min(r_1, r_2)^\lambda}{\max(r_1, r_2)^{\lambda+1}}.$$

We assume, it is sufficient to focus on diagonals when discretizing these functions and pick a local basis f_i localized around r_i (for example DVR polynomials). Then assuming $|r_i - r_j| \gg 0$, we get

$$[D_\lambda]_{ij} \approx \frac{4\pi}{2\lambda + 1} \frac{\min(r_i, r_j)^\lambda}{\max(r_i, r_j)^{\lambda+1}} \approx \frac{1}{2\lambda + 1} \left(\frac{\min(r_1, r_2)}{\max(r_1, r_2)} \right)^\lambda [D_0]_{ij}.$$

Where both electrons are well-separated, the multipole expansion coefficients are exponentially suppressed and one should be able to truncate the sum at quite small Λ . The effects of this truncation is to bring $F1$ closer to $F2$ apply count-wise.

2.4. Constrained Basis

One of the facts that makes calculations with angular momenta high enough even possible is that for the physical situations we are interested in there are usually some constraints we can make use of. In first iteration, we mean by constraint a conserved quantity. More realistically, constraints appear even if symmetries are realized only approximately.

The following symmetries are of practical importance in the situations under investigation:

1. In the case of a linearly polarized Laser, the Hamiltonian has a $SO(2)$ symmetry around the polarization axis z ; this implies an exact conservation of $m+n = M$. The initial state satisfies $M = 0$, and we can restrict the basis to $m+n = 0$ without making errors.

2. In the case of a circularly polarized Laser, the conserved charges are given by $l - m = 0$ and $p - n = 0$. As the situation we are dealing with is a Laser pulse with a smooth envelope, the Laser is not circularly polarized (there is a direction, where the amplitude reaches its maximum). However, the conservation still holds approximately and limiting the basis to $l - m < C_{lm}$, $p - n < C_{lm}$ gives converged results [4].

In addition to this, one finds in practice [48] that not both electrons attain high angular momenta at the same time. One can limit the basis to $l < C_u \vee p < C_u$. These constraints are quite strong; in the results presented in the next section, we can usually do with $C_{lm} = 3$, $C_u = 3$.

Since this text deals with circular polarization, in the following we will only be concerned with imposing both C_{lm} and C_u at the same time.

2.4.1. Explicit Schemes

Fig. 2.8 gives the application counts of the schemes $F1$ and $F2$ under the constraints $C_u = C_{lm} = 3$. We see that both schemes still scale better than the full matrix, as the condition $m - m' = n' - n$ holds. Furthermore, as we imposed three constraints, the scaling of the basis size approaches 1 (and therefore the full matrix size scales roughly like L^2).

2.4.2. Product Scheme

As in the unconstrained case the scheme P beats $F1$ only by a narrow margin and the constraint C_u only decreases improvements stemming from tensor product schemes, we do not consider P in the setting of heavily constrained bases.

2.4.3. Grid Transformations

As we have seen, employing both constraints C_u and C_{lm} effectively reduces the scaling of basis size to linear. However, when transforming to a grid (we restrict ourselves to $G2$ here) we still need to span the whole space of P_l^m for $l < L$ and $-C_{lm}/2 < m \leq l$. This means, the η grid will still need to scale like κL for each electron; the product grid will therefore scale like L^2 . We expect the application of the diagonal potential to scale worse than the schemes $F1$ and $F2$ (both around $L^{5/4}$ in the special case presented above).

Let us investigate App_{G2} nonetheless. In step one and four, the η transformations, the right-hand-side indices of T_m (left-hand-side indices of T_m^\dagger) can be limited to include only admissible angular momenta. In step three, the application of the diagonals, we are restricted to admissible values of m , n , m' and n' . Numerically carrying out the summation gives us the application counts presented in Fig. 2.9; we see that both grids perform worse than a full matrix for $L \gtrsim 20$. Hence, we do not consider grid transformations in the case of (strongly) constrained bases.

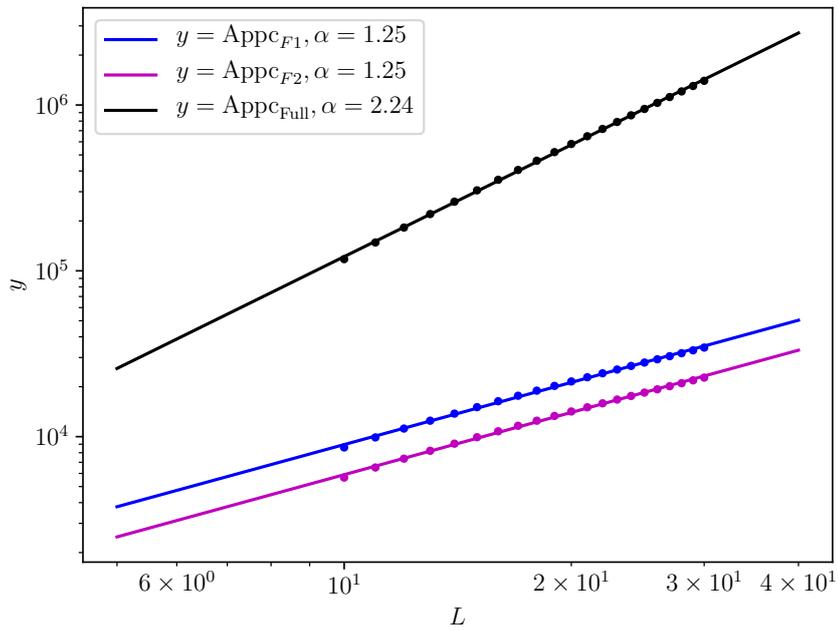


Figure 2.8.: Applications count of the schemes $F1$ and $F2$ under the constraints $C_u = C_{lm} = 3$. “Full” is given by the square of the constrained basis size.

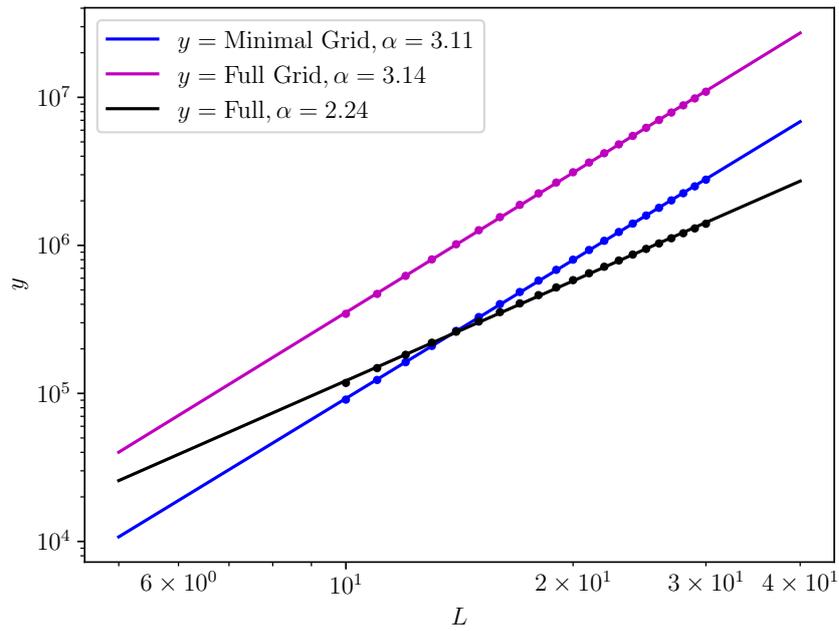


Figure 2.9.: Applications count of the scheme $G2$ for both full and minimal grids under the constraints $C_U = C_{lm} = 3$. “Full” is given by the square of the constrained basis size.

2.4.4. Coupled Angular Momenta

When transforming from the basis $lmpn$ to $lpJM$, the constraint C_U can be implemented in the same manner. However, the constraint C_{lm} produces bloating of the basis if we require the transformation T from the product basis to the coupled basis to be unitary $T^\dagger T = \text{Id}$. By bloating we mean the basis of coupled angular momenta is larger than the product basis.

Let us investigate this in more detail. For the transformation to be unitary, we need the coupled basis to include every J, M for which there exist l, p, m and n satisfying all constraints and

$$\langle lm, pn | lp, JM \rangle \neq 0.$$

We can make use of the selection rules to derive the condition

$$l - m < C_{lm} \wedge p - n < C_{lm} \Rightarrow l + p - M < 2C_{lm} \Rightarrow J - M < 2C_{lm}.$$

This implies, we need to impose $l + p - M < 2C_{lm}$ in order to cover the whole product basis. The applications count under this constraint is compared to the full matrix in $lpmn|p'm'n'$ under the constraints $C_U = C_{lm} = 3$ in Fig. 2.10.

Whether this is necessary, is not clear a priori. Rather, we presume the constraint $J - M < 2C_{lm}$ is chosen unnecessarily weak when propagating in the coupled basis. To investigate this, we propagate a sample situation in the product basis utilizing both constraints. We then apply the minimal unitary transformation at certain timesteps and look at the resulting coefficients. Fig. 2.11 gives a comparison of the distribution of the probability over the angular coefficients. Plotted are maxima over time of the normsquares of the subvectors for given angular indices. This numerical experiment suggests, it is sufficient to impose the constraint $J - M < C_{lm}$. For the sake of completeness, Fig. 2.12 gives the application counts under this assumption.

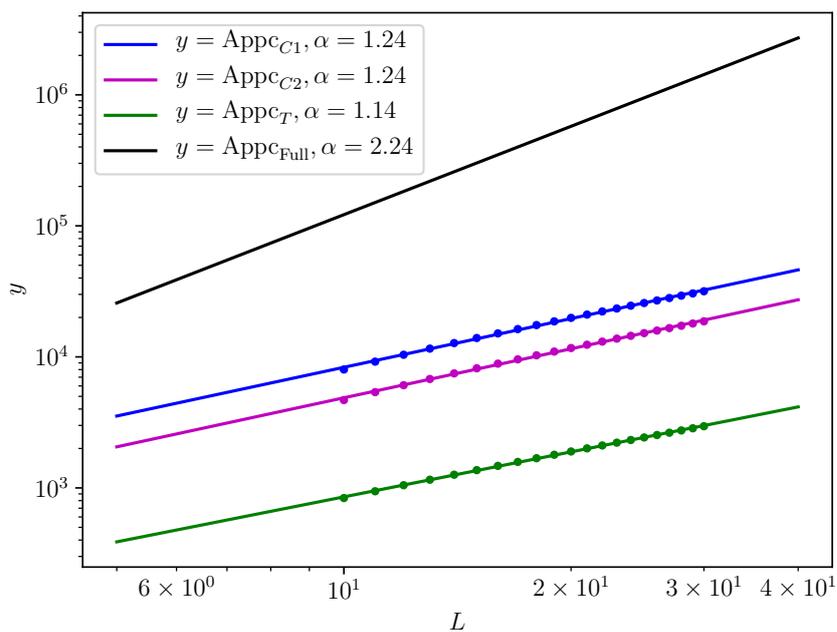


Figure 2.10.: Applications count of the schemes $C1$ and $C2$ under the constraints $C_u = 3$ and $l + p - M < 6$, covering the whole constrained product basis with $C_u = C_{lm} = 3$. “Full” is given by the square of the constrained product basis size.

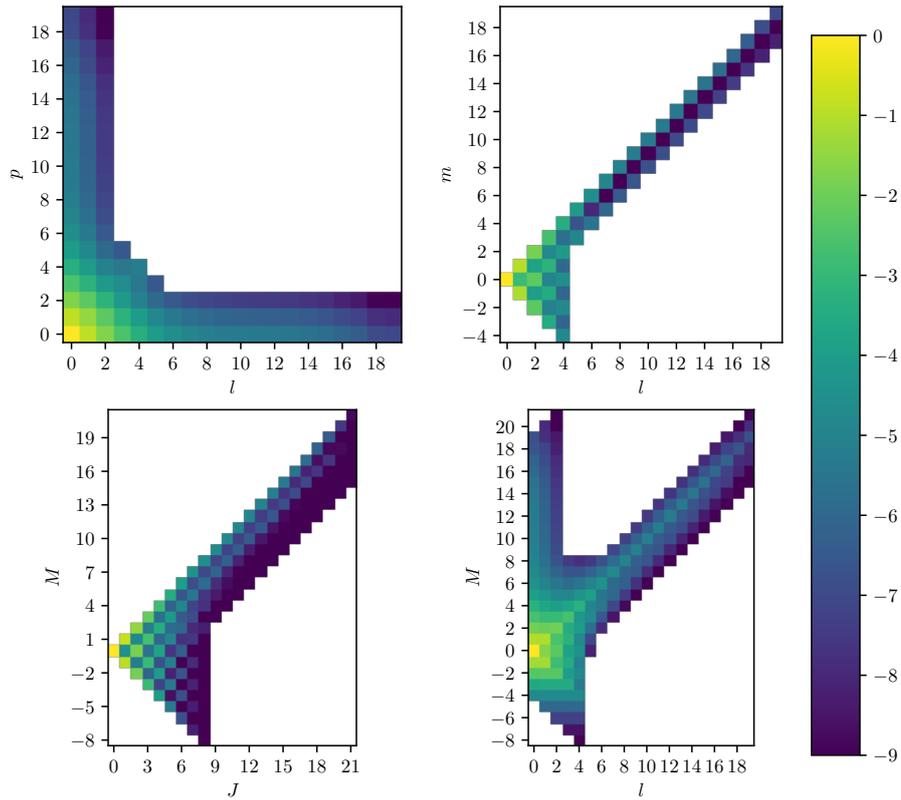


Figure 2.11.: Angular distribution of the probability during time propagation of an example situation. Plotted are maxima over time. Constraints imposed are $C_{lm} = C_{ll} = 3$. White regions are excluded by constraints.

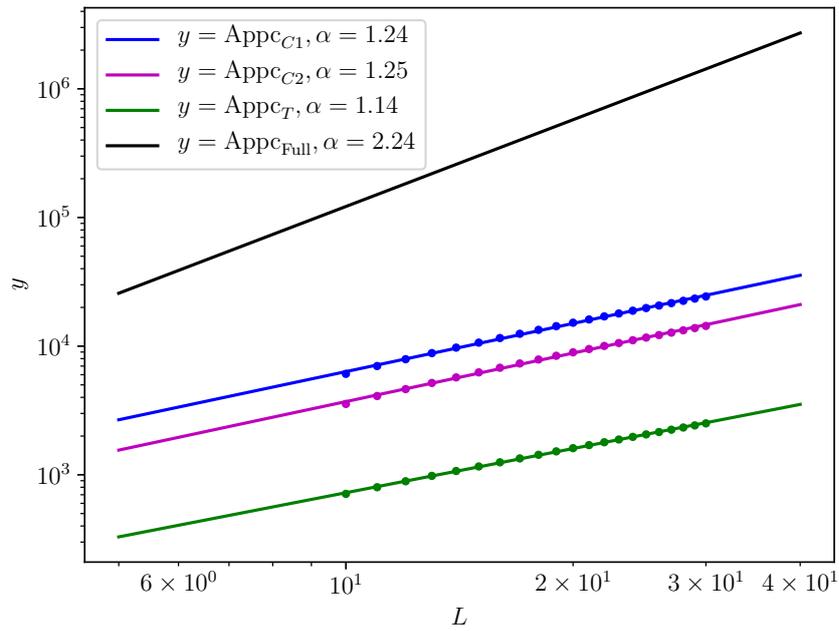


Figure 2.12.: Applications count of the schemes $C1$ and $C2$ under the constraints $C_U = 3$ and $J - M < 3$. “Full” is given by the square of the constrained product basis size.

2.5. Conclusion

At this point, we can draw conclusions for the two cases of constrained and full basis. The two situations are plotted in Figs. 2.13 and 2.14.

If the basis is not constrained, grid transformations will sooner or later be the fastest schemes. Both $G1$ and $G2$ exhibit L^5 scaling, whereas full schemes or coupled angular momenta scale like L^7 resp. L^6 . However, the tensor product structure of the grid transformations implies a high prefactor, whereas full/coupled schemes contain quite sparse matrices. In the end, even taking into account additional penalties in the schemes $F1$ and $C1$ arising from the need to calculate A_λ matrix elements, it appears unlikely that grid transformations will yield a significant speedup in the L range that is currently feasible in `tRecX`. The better scaling of $C1$, $C2$ compared to $F1$ and $F2$ suggest, there is to be gained when moving from a product basis to coupled angular momenta. The product scheme P eventually beats the full schemes $F1$; however, only by a small margin. We consider it a more fruitful approach to implement a coupled scheme or a grid transformation instead of the product P .

In the case of a heavily constrained basis both full and coupled scheme produce the same scaling; grid transformations inherently cannot profit from constraints and scale worse. The prefactors of the C and F schemes tend to favour coupled bases if equally strong constraints can be chosen in JM . However, the exact prefactors do depend on actual implementations; so, these results do not suggest it is strongly beneficial to move from the product to a coupled basis. The product scheme P has not been considered in this setting.

In practice, one modulates between both situations as for example the constraint C_{ll} cannot be imposed this strictly. Instead one needs to enlarge the basis for small l to allow for correlations near the origin (see Chapter 3).

It is important to point out that our results are only a first step in the direction of investigating these schemes in full detail. Most importantly, the question “how well do these schemes parallelize?” has not been asked nor answered. Data-locality, an important factor in the performance of any algorithm, has not been considered. Furthermore, implementation-dependent penalties, as e. g. the calculation of Gaunt coefficients from quadrature, have not been taken into account.

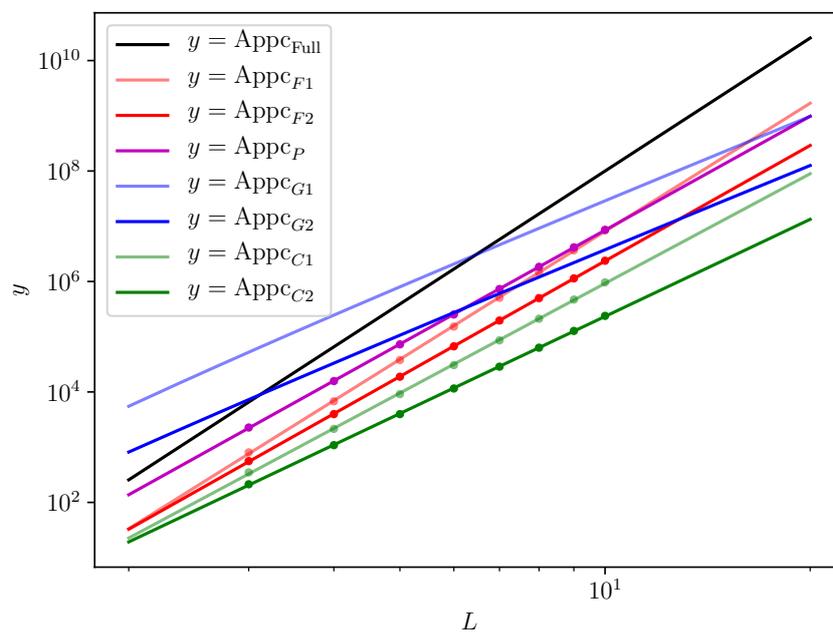


Figure 2.13.: Comparison of different application schemes for the full basis L^4

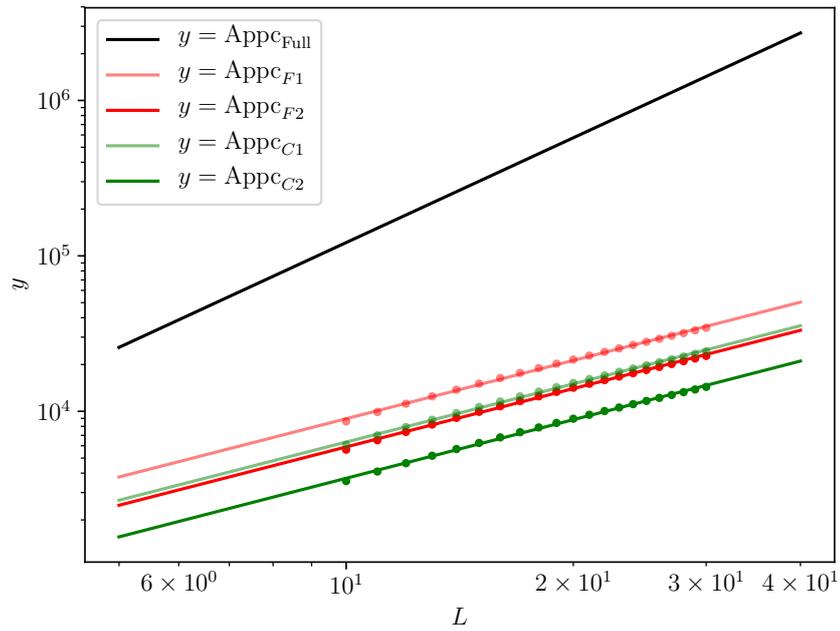


Figure 2.14.: Comparison of different application schemes for the constrained basis: $C_{ll} = C_{lm} = 3$ in the product basis case, $C_{ll} = 3$, $J - M < 3$ in the coupled case.

3. Helium in an Ultrashort Laser Pulse

3.1. Physical Motivation

3.1.1. The Attosecond Clock

In 2014 it became possible to measure the angular distribution of electron emission from a Helium gas subjected to an IR elliptically polarized ultrashort Laser to a high accuracy [1]. The results were interpreted as a measure of the time it takes an electron to tunnel through the potential barrier; we will elaborate on this in a second. The interpretation of this result as well as of the experimental data, however, remain debated to this day [2–4].

Throughout this part we use atomic units $e^2 = m_e = \hbar = 1$. To start off the discussion, we note that there are two main processes contributing to ionization in this regime. The potential barrier is drawn in Fig. 3.1. A bound electron subjected to the Laser field can either pick up many photons to escape the barrier through the vertical path (multi-photon ionization), which essentially corresponds to the photoelectric effect. Alternatively, if the field is strong enough, the potential barrier is thin enough for tunnel ionization (horizontal path) to play a significant role. These two processes are characterized by the *Keldysh parameter* [50]

$$\gamma = \sqrt{\frac{I_p}{2U_p}} = \frac{\omega\sqrt{2I_p}}{E},$$

with the ionization potential I_p , photon energy ω , and the peak electric field E . U_p denotes the *ponderomotive potential* of the pulse, the mean kinetic energy of a classical electron in a plane-wave electromagnetic field

$$U_p = \frac{E^2}{4\omega^2}.$$

As γ increases, multi-photon ionization becomes more likely. In the domain $\gamma \ll 1$, tunnel ionization is the main factor, whereas in the cross-over regime $\gamma \approx 1$ both effects contribute. The experiments were conducted with $\gamma \approx 1$ showing both effects with a focus on tunnel ionization.

3.1.2. Classical Picture

To motivate how this experiment can be linked to tunneling time, we describe the situation in a simplified manner (for this calculation as well as less simplified analyses we refer the reader to [2]). Assume a pulse given by the field strength

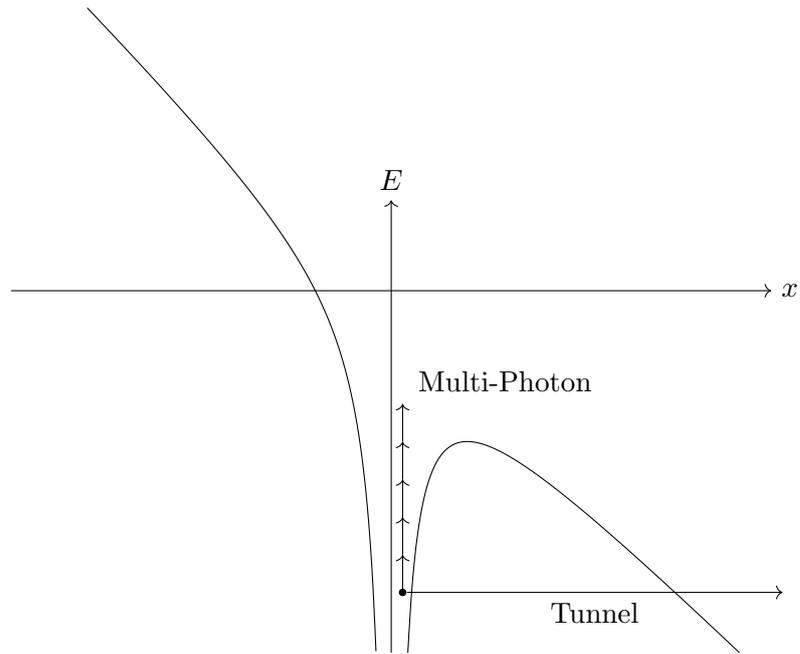


Figure 3.1.: The two ionization channels of a bound electron subjected to a strong Laser field. The vertical path represents multi-photon ionization, the horizontal path tunnel ionization. The electric field is oriented along the negative x axis. E is the electron energy.

$$\vec{E}(t) = f(t) \begin{pmatrix} -\sin \omega t \\ \cos \omega t \\ 0 \end{pmatrix},$$

where $f(t)$ describes the envelope of the pulse reaching its maximum at $t = 0$. This pulse rotates in mathematically positive direction in the xy plane. Let us simplify the ionization process crudely. We assume that at $t = 0$, where the field reaches its maximum strength, the electron tunnels instantaneously through the barrier created by the Coulomb potential and the Laser field; afterwards, the electron has zero momentum [2] and is accelerated by the Laser field, which we assume to be much stronger than the Coulomb attraction of the ion. Therefore, we neglect the Coulomb attraction.

The direction in which the electron escapes can be calculated classically by noticing that the canonical momentum

$$\pi(t) = m\vec{v}(t) - e\vec{A}(t),$$

with the electron velocity \vec{v} and the vector potential \vec{A} , is a conserved quantity. The electron escapes at $t = 0$; denote the time it is detected as $t = T$. Then

$$\begin{aligned} \pi(T) &= m\vec{v}(T) - e\vec{A}(T) = m\vec{v}(0) - e\vec{A}(0) = \pi(0) \\ \Rightarrow m\vec{v}(T) &= -e\vec{A}(0), \end{aligned}$$

assuming the pulse has finished by T . Up to effects of the pulse envelope, the electric field at $t \approx 0$ can be generated by

$$\vec{A}(t) = -\frac{f(0)}{\omega} \begin{pmatrix} \cos \omega t \\ \sin \omega t \\ 0 \end{pmatrix}.$$

Therefore, the electron escapes in positive x direction, $\phi = 0$. This situation is visualized in Fig. 3.2. From this, we see that the neglected effects of the Coulomb force provide a distortion towards positive ϕ angles. The interpretation of the experiment compensates for this factor using a semi-classical simulation [1, 2].

3.1.3. Experimental and Numerical Results

Landsman and colleagues [1] found that the deflection angle, the angle ϕ of maximal emission, exceeds the values expected from Coulomb corrections significantly. This result was interpreted as measuring the time it takes the electron to tunnel through the potential barrier as during that time the angle of the electric field moves on towards higher ϕ . (The Laser field is compared to the hand of an *attosecond clock*.)

This interpretation, however, remains debated as it is yet to be confirmed numerically. In 2015 it was found, employing different numerical techniques, among them

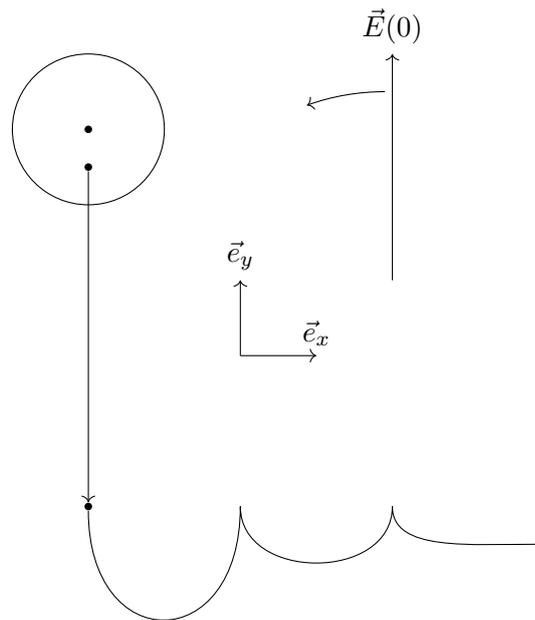


Figure 3.2.: Visualization of the classical interpretation of ionization. Ionization happens through instant tunneling at $t = 0$ after which the electron propagates classically in the Laser field. In this picture it is clear that the Coulomb tail provides a deflection in the positive ϕ direction.

tSURFF,¹ that for Hydrogen the deflection angle can be explained entirely in terms of the Coulomb deflection [3]. To this end, in addition to Hydrogen calculations, simulations with a Yukawa potential instead of the Coulomb potential were conducted. This potential resembles the Coulomb potential near the nucleus and drops off exponentially for larger distances. This means, the potential barrier to be tunneled through is largely unchanged, whereas deflection effects of the long-range parts are suppressed. A deflection angle of zero was found in this case.

This quite convincingly disproves the idea of tunneling time; however, the measured deflection angle may still be a result of electron-electron correlation effects during tunneling. In 2017 numerical simulations employing the haCC technique, where one electron moves freely and one is constrained to the bound spectrum of the Coulomb potential, found no impacts of correlation on the deflection angle [4]. In particular, the deflection can still be explained in terms of Coulomb deflection. To quote the authors

The only remaining loophole on the theory side could be closed by including the full double-electron continuum. Although such a calculation is feasible using tSurff and preponderance rules, we do not consider this a fruitful pursuit. Rather, it may be worth while to re-examine the match between experimental setting and theoretical modeling, or, possibly, further simplify the physical situation with experiments on atomic hydrogen.

As the discrepancy between theoretical (numerical) and experimental results continues to exist, we will investigate the impact of fully-modelled two-electron correlations nonetheless. This is the goal of this part. We will, however, not work with the parameters matching the experiment as an IR calculation currently takes too much time. Instead, we use a 400nm circularly polarized pulse. The aim is to discern whether a full Helium calculation yields different deflection angles from a one-electron calculation. If this is not the case, it appears less and less likely that a calculation in the experimental setting gives deflection angles exceeding Coulomb deflection.

3.2. Technology

3.2.1. tSURFF

Calculation of photo-electron spectra in fully-correlated multi-electron systems (as in this case Helium) is made possible by the *time-dependent surface flux* (tSURFF) method [51, 52]. This method exploits the intuition that complicated dynamics happen near the nucleus, whereas the dynamics far outside are of an easier nature and can be handled analytically. We will now give a short summary of the workings; for a full derivation and numerical proofs we refer the reader to the above citations.

Without tSURFF, emission spectra are calculated by propagating the wave-function under the full Hamiltonian (in our case Helium plus dipole Laser field) until a time T

¹see Section 3.2.1

after the pulse has finished and ionized electrons move freely away from the nucleus. One then decomposes this final amplitude using the scattering solutions of the field-free Hamiltonian. This requires knowledge of the final wave-function over a large (thousands of atomic units) spatial extent; furthermore, the discretization must be able to represent the characteristic momenta of the photo-electrons; therefore, the number of grid points per atomic unit is bound from below. In the end, spatial discretizations require in the order of thousands of grid points per spatial dimension. Additionally, calculation of the scattering solutions is a computationally hard task.

tsURFF starts from the assumption that outside of a certain radius R_c the field-free Hamiltonian is essentially given by the Laplacian. (This assumption can be modified to equate the Hamiltonian outside of R_c with a Hydrogen-like atom [51]; however, such an extension is currently not implemented within the framework used in this text.) Furthermore, we assume that the part of the bound spectrum which is important for the process under investigation has negligible probability outside of R_c . These assumptions essentially are the ones defining scattering theory; however, the important realization is that such an R_c can be of microscopic order (one hundred atomic units or even twenty atomic units). This means, we can split the wave-function at time T (for a one-electron system)

$$\psi(\vec{r}, T) = \psi_B(\vec{r}, T) + \psi_S(\vec{r}, T),$$

with ψ_B in the bound spectrum of the field-free Hamiltonian, essentially within $[0, R_c]$ (This does not hold in a strict mathematical sense as, for example, the eigenstates of the Hydrogen atom decay exponentially, but they are supported on all of \mathbb{R}^3).

Under the above assumptions one can show [51] that the wave-function ψ_S outside of R_c is determined solely by the values and derivatives of the wave-function at R_c at all times during propagation. These surface values can be obtained efficiently if we are able to implement a perfectly absorbing boundary at $R \geq R_c$ (see next section). We can therefore propagate the wave-function on a spatial discretization $[0, R]$ capturing the surface values at R_c and then use these in a second step to calculate the wave-function on $[R_c, \infty]$.

To calculate the wave-function ψ_S , one requires analytical knowledge of the eigenstates of the Hamiltonian outside of R_c , the kinetic energy plus the Laser field Hamiltonian; these are known as the *Volkov solutions* labelled by momentum \vec{k} [51]

$$\chi_{\vec{k}}(\vec{r}, t) = (2\pi)^{-3/2} \exp \left[i\vec{k}\vec{r} - \frac{i}{2} \int_{-\infty}^t (\vec{k} - \vec{A}(\tau))^2 d\tau \right].$$

Therefore, the application of the Hamiltonian in the time evolution outside of R_c reduces to the application of the Volkov phase, diagonal in the momentum basis. The fully-differential spectrum can then be readily obtained from the amplitude ψ_S in momentum basis.

This method extends to multi-electron systems in a natural manner [52]. There is, however, one additional implication of the assumption; the Coulomb interaction between the electrons is switched off outside R_c ; therefore, double ionization spectra

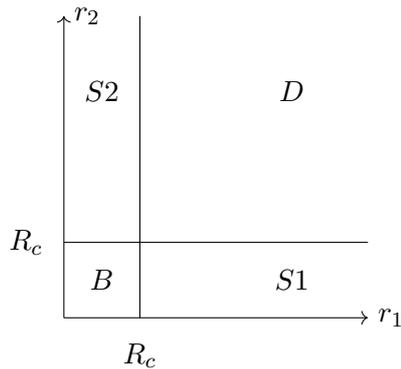


Figure 3.3.: Splitting of the wave-function into bound, singly-ionized and doubly ionized parts employing \mathfrak{tSURFF} for a two-electron system. r_1 and r_2 are the radii in spherical coordinates of the two electrons; R_c is the \mathfrak{tSURFF} radius

are distorted along parallel emission directions. This problem is of a deep nature and troubles scattering theory of multi-ionization processes. However, this text is concerned only with single-ionization processes, where the implications of the cutoff radius R_c are, as in the single-electron case, the missing Coulomb tail. Fig. 3.3 gives the splitting of the Helium wave function into four parts: the bound part B , the single-ionization parts $S1$ and $S2$ and the double ionized wave function in D . Exchange symmetry dictates equivalence of the spectra obtained from $S1$ and $S2$; therefore the single-ionization spectrum can be obtained from propagation in the bound region B followed by $S1$ or $S2$ propagation, where one electron is subjected to the Hydrogen-like Hamiltonian of the Helium ion and the other is subjected to the Volkov phase. Fully-differential and channel-resolved (where the *channel* c labels the bound state occupied by the electron in the ion) can be obtained readily from storing the amplitude (in momentum basis) $\psi_{S2}(c, k_2, \eta_2, \phi_2)$ during propagation.

3.2.2. irECS

The absorbing boundaries required for efficient use of \mathfrak{tSURFF} are implemented using *infinite-range exterior complex scaling* (**irECS**) [53]. **irECS** is a numerically very efficient implementation of the mathematical concept of exterior complex scaling [54]. The headline is that **irECS** can be considered a *perfect absorber*, i. e. the reflections generated by **irECS** are below machine precision [53]. We will now give a short motivation of why exterior complex scaling can be used to implement absorbing boundaries and its implementation using infinite-range bases. For a detailed discussion, we refer the reader to [53] and references therein.

Let us focus on $L^2(\mathbb{R})$ and define exterior real scaling for $\psi \in L^2(\mathbb{R})$ as

$$(U_\lambda \psi)(x) = \begin{cases} \psi(x), & |x| < R \\ e^{\lambda/2} \psi(\text{sign}(x)(e^\lambda(|x| - R) + R)), & \text{else,} \end{cases}$$

for scaling radius R and factor $\lambda \in \mathbb{R}$. This unitary transformation implements scaling of the basis outside of R . Since U_λ is unitary, we can define the scaled Hamiltonian ($H : L^2(\mathbb{R}) \rightarrow L^2(\mathbb{R})$) as

$$H_\lambda = U_\lambda H U_\lambda^\dagger,$$

with the same spectrum as H . For a certain class of potentials V , called *dilation analytic*, the mapping from $\lambda \in \mathbb{R}$ to $(-\Delta + V)_\lambda$ is analytic and therefore can be extended uniquely to the complex plane $\lambda \in \mathbb{C}$. The resulting Hamiltonians are no longer self-adjoint; however, one can show, for dilation analytic potentials, that the bound spectrum remains unchanged, whereas the continuous spectrum is rotated into the complex plane as shown in Fig. 3.4. The implication of this can be seen from replacing ω by $\omega - i\gamma$ in a plane wave corresponding to the continuous spectrum of $-\Delta$

$$\psi_k(x, t) = \exp[i(kx - \omega t)] \rightarrow \exp[i(kx - \omega t) - \gamma t],$$

i. e. the wave is exponentially damped. This motivates that ECS is able to implement absorption of outgoing waves; to do this, one replaces time evolution by H with time-evolution by H_λ . This non-Hermitian operator generates a non-unitary evolution, where outgoing waves are absorbed at R . In practice, one replaces the grid on \mathbb{R} by a grid on $S_{R,\lambda}(\mathbb{R})$:

$$S_{R,\lambda}(x) = \begin{cases} x, & |x| < R \\ \text{sign}(x)(e^\lambda(|x| - R) + R), & \text{else.} \end{cases}$$

With this we can see the damping of outgoing waves by replacing x with $S_{R,\lambda}(x)$; for $|x| \geq R$ we have

$$\psi_k(x, t) = \exp[i(kx - \omega t)] \rightarrow \exp[i(kS_{R,\lambda}(x) - \omega t)] \propto \exp\left[-k\text{Im}\left(e^\lambda\right)x\right].$$

Originally this method was used by limiting absorption to a certain interval $[R, R_1]$. To achieve perfect absorption, **irECS** replaces this finite interval by infinite range basis functions of polynomial times $\exp[-\alpha|x|]$. Using this, absorption up to machine precision happens with as few as ten to twenty coefficients in this interval.

3.2.3. tRecX

The **tRecX**² software package, created by Armin Scrinzi, implements the heavy-lifting associated with solving initial-value problems, such as the Schrödinger equation. **tRecX** is implemented in C++. The name stems from

²<https://trechx.physik.lmu.de/home.html>

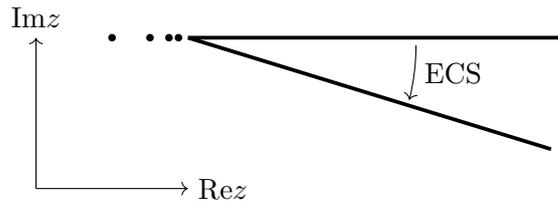


Figure 3.4.: Spectrum of a Schrödinger Hamiltonian under exterior complex scaling $\sigma [(-\Delta + V)_\lambda]$. Dots represent point spectrum, which remains unchanged. The lines represent the continuous spectrum, which is rotated depending on $\text{Im}(e^\lambda) \lambda$.

$$\text{tRecX} = \begin{cases} \text{tSURFF} + \text{irECS} \\ \text{t(ime dependent) rec(ursive inde)x(ing)}. \end{cases}$$

Recursive Indexing The problem solved by `tRecX` is the efficient handling of multi-index quantities, such as wave-functions and operators. This is achieved using *recursive indexing*, i.e. implementing the multi-index as a recursive data structure, a tree. Let us focus on the three-dimensional case with the multi-index ψ_{lmix} , where l is angular momentum, m is magnetic momentum, i labels the finite element and x the grid point within that finite element. This index structure is implemented as a tree, where every node corresponds to a sub-vector; the root corresponds to the whole vector and leaves correspond to scalars. Upon traversing the tree from top to bottom, one picks up specific values for the indices l , m , and so on, step-by-step refining the sub-vector. This *index tree* is exemplified in Fig. 3.5. In this figure, the first child of the first child of the root corresponds to the sub-vector of fixed $l = m = 0$.

This index structure lies at the heart of `tRecX` and defines identity of indices; by that we mean that a pointer to some node in the index hierarchy uniquely defines the sub-vector of a wave-function or the space on which an operator block acts. The recursive structure has proven to be extremely versatile in that it elegantly allows the implementation of dependencies between indices (as for example $|m| < l$ or the preponderance rule). Furthermore, many complicated operations can be brought to quite simple form when expressed recursively; as an example, permutation of the tree (that is, exchanging of layers, which corresponds to permutation of the vector) is implemented in less than thirty lines of code.³

For every discretization (finite-dimensional vector space), there exists one such index structure in the program memory; these vector spaces are, for example, the state space (Hilbert space) or the spectral space of an operator (where it acts as a diagonal matrix).

³see the class `Tree<T>`

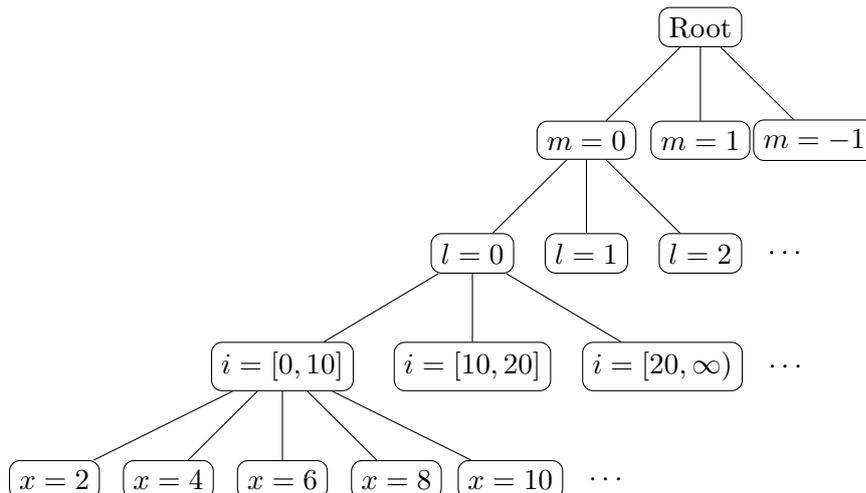


Figure 3.5.: The index tree at the core of `tRecX`. Only children of the first child are drawn at each level; traversing the tree from top to bottom, one refines the subvector arriving at the scalar $m = l = 0$, $x = 2$ for the left-most leaf.

Handling of Vectors and Operators Vectors and Operators are represented as a tree matching the structure of the index. However, whereas the index structure continues down to leaves corresponding to scalars, vector and operator trees only go down to the so-called *floor level*, such that leaves correspond to sub-vectors and operator blocks. This is exemplified in Fig. 3.6; the vector/operator tree leaves (the so-called floors) contain the actual data. Every node in a vector tree has its index defined by a pointer to a node in the index structure, whereas a node in an operator tree has two pointers, one for the left-hand-side index and one for the right-hand-side one. This structure enables setup as well as application of operators by traversing from the root downward. Setup can be handled efficiently as for example selection rules in the electron-electron interaction can be implemented on the top levels without need to traverse downward in blocks that end up to be zero. With vector space dimensions of up to 10^7 , this constitutes a huge improvement in setup speed as well as memory.

Furthermore, the operator leaves (corresponding to matrix blocks) are implemented as an abstract class, allowing for efficient adaption to situations like diagonal blocks or tensor products. Structures like these can be enforced as well as automatically detected during setup. As we have seen, the electron-electron interaction cannot be stored as a full matrix, even making use of selection rules, due to memory limitations. Instead this implementation allows for a customized operator tree floor implementing the application through multipole expansion.

An Example: Product Projections `tRecX` uses an explicit Runge-Kutta scheme for propagation; the largest possible step-size of such a scheme is proportional to the inverse of the largest eigenvalue of the generator of time evolution. The kinetic

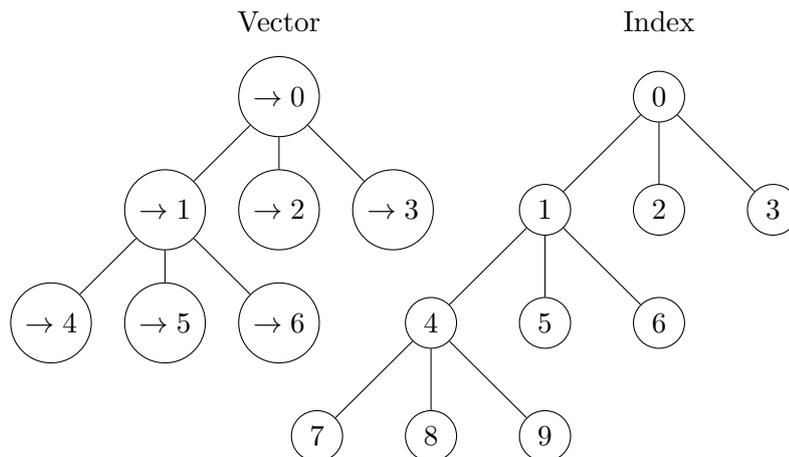


Figure 3.6.: Vector and index tree structures side-by-side. Numbers on index nodes represent memory addresses; arrows on the vector nodes represent pointer to the memory location, giving the sub-vectors their index. The vector tree structure only goes down to the floor level, where leaves correspond to sub-vectors (in this case, the left-most leaf of the vector tree corresponds to a sub-vector of size three) and contain the actual data.

energy operator leads to a *stiff* problem, meaning the largest eigenvalue is very high and therefore the largest step-size possible very small.

However, as it turns out in practice, high eigenvalues are not needed, i. e. one can obtain converged results by propagating with PHP instead of H , where $P^2 = P^\dagger = P$ projects out eigenvalues of the Hamiltonian larger than E_{cut} (see Section 3.3).

In the Helium case, one finds that a speedup is achieved, even if we do not project out high eigenvalues of the full Hamiltonian but eigenvalues of $H_1 \otimes \text{Id} + \text{Id} \otimes H_1$, where H_1 is the Helium ion (Hydrogen-like) Hamiltonian. In other words, we can leave out the electron-electron interaction in the projection and still achieve significant speedups. The important gain from working without the electron-electron interaction stems from the fact that the high energy-projector has tensor product structure

$$P = P_1 \otimes P_1 = (\text{Id} \otimes P_1)(P_1 \otimes \text{Id}).$$

More precisely, if P_1 projects onto the eigenvalue range $(-\infty, E_{\text{cut}}/2]$, $P_1 \otimes P_1$ acts as identity on the eigenvalue range $(-\infty, E_{\text{cut}}/2]$ and projects out eigenvalues larger than E_{cut} .

These tensor products $P_1 \otimes \text{Id}$ can be implemented in a straight-forward manner given the index tree structure and P_1 . By traversing the index structure on both sides of the projection simultaneously, one can easily discern diagonal levels and build the tensor product of P_1 with the identity.⁴ This construction easily generalizes to cases

⁴See the classes `DiscretizationSpectralProduct` and `TensorOperatorTreeWithId`.

where it is not a tensor product in the strict sense as, for example, the constraint C_U is imposed.

Outlook This section gave a quick introduction into the core concepts of `tRecX`; the software itself has grown to a big project⁵ and giving a detailed overview over all functions is outside of the scope of this text. Just to mention a few:

- The definition of the physical situation (discretization, constraints, Laser, and so on) is handled through a versatile scripting language, which allows `tRecX` to be used without knowledge of `C++`.
- Large-scale parallelization is handled via MPI (The results presented later were obtained using the parallelized code.)
- During propagation, operator blocks that do not contribute are automatically detected and switched off.
- Various spectra obtained from the fully-differential result can be plotted without any additional scripts.
- Timing and memory consumption can be analyzed to any degree of detail by using preprocessor macros, which are switched off in production runs to not generate additional overhead.
- `irECS` is built naturally into the handling of discretizations and basis sets.
- `tSURFF` is implemented and has been tested extensively.

3.3. Results

All results presented here have been obtained using the `tRecX` software package.

3.3.1. Observables and Convergence

As we cannot run a simulation on the infinite-dimensional Hilbert space of the two electrons in a Helium atom, we need to reason about *convergence*. The idea is to describe the discretization we are using with a set of parameters and obtain a value for every observable of interest for some combination of these parameters. The actual value of the observable is then given (up to machine precision) by the limit as all parameters tend to infinity (parameters like basis size) or zero (parameters like accuracy).⁶

The first iteration of this idea is to compute the value for a given set of parameters, then increase each parameter significantly (for example, multiply the radial basis

⁵The `clloc` utility (<https://github.com/AlDanial/clloc>) finds around 55000 lines of `C++` code.

⁶We will always use the term “increase” to mean “better,” i. e. an accuracy of 10^{-7} can be increased to an accuracy of 10^{-8}

size by ten) and compute the new value of the observable; if the discrepancy is below machine precision, the result is considered converged. We usually change this criterion to requiring no change in the result if any *one* parameter is increased.

This convergence notion in the strong sense can be employed in many situations; however, simulating the full two-electron Helium requires weakening the criterion as e. g. a radial basis size ten times as large is no longer feasible. This means, we weaken the strength of our results by requiring them to be only converged to a few percent or ten percent upon increasing the basis size by a factor of 1.5. These notions, however, become less and less rigorous; so at this point it is important to *define* the observables of interest and what we mean by *converged results*.

The observable, we focus on, is the fully-differential single-ionization spectrum σ obtained from the amplitude ψ in either subregion (i. e. we do not calculate ionization of the first as well as the second electron, but exploit the symmetry of the wavefunction):

$$\sigma(c, k, \eta, \phi) = 2|\psi_{S_2}(c, k, \eta, \phi)|^2,$$

where k , η and ϕ are the polar coordinates of the ionized electron and c labels the bound state of the electron remaining in the ion (the channel); that is, $c = 1s, 2s, 2p$ and so on. This spectrum is normalized such that the total yield (probability of single-ionization of an electron) p is given by

$$p = \sum_c \int_0^\infty dk \int_{-1}^1 d\eta \int_0^{2\pi} d\phi k^2 \sigma(c, k, \eta, \phi).$$

At this point, we are able to define convergence by requiring that σ at any grid point does not change by more than one percent if any parameter is increased significantly. However, this is still too strong and unpractical a requirement as the fully-differential spectrum contains a huge amount of information that we are not interested in.

Let us take a step back and reason about the observables we need. As we have seen earlier, the key motivation for this simulation is the deflection angle of the maximal yield in the xy plane. Furthermore, the momentum spectrum gives useful and quick insights into which parameters need to be chosen how high. Therefore, we define the two observables of interest as:

1. The *total spectrum*

$$\sigma(k) = \sum_c \int_{-1}^1 d\eta \int_0^{2\pi} d\phi \sigma(c, k, \eta, \phi),$$

or, more practical, energy resolved

$$\sigma(E) = \sqrt{2E} \sum_c \int_{-1}^1 d\eta \int_0^{2\pi} d\phi \sigma(c, \sqrt{2E}, \eta, \phi).$$

The prefactor stems from the normalization

$$\int_0^\infty dk k^2 \sigma(k) = \int_0^\infty dE \sigma(E).$$

We limit the spectrum to the range $\sigma \geq 0.01 \max_E \sigma(E)$.

2. Maxima and minima of the ϕ resolved spectrum in the xy plane

$$\sigma_{xy}(\phi) = \alpha \sum_c \int_0^\infty dk k^2 \sigma(c, k, 0, \phi),$$

where α is defined such that $\max_\phi \sigma_{xy}(\phi) = 1$. To make this observable more useful, we adapt the definition to a hypothetical experimental situation. Suppose the contrast between maximum and minimum can be resolved in 100 steps. Define

$$\sigma_{xy}^{\max/\min} = \max / \min_{\phi \in [0, 2\pi]} \sigma_{xy}(\phi).$$

Now define the regions Φ_{\max} and Φ_{\min} as the sets of ϕ for which $\sigma_{xy}(\phi)$ exceeds $0.99 \times \sigma_{xy}^{\max}$ respectively lies below $1.01 \times \sigma_{xy}^{\min}$. These regions represent the best resolution of angular maxima/minima possible in an experiment.

We characterize these regions by $\phi_{\max/\min}$, the center points of the intervals and $\Delta\phi_{\max/\min}$, the width of the intervals.

Having defined the observables $\sigma(E)$, $\phi_{\min/\max}$ and $\Delta\phi_{\min/\max}$ we now need to make clear what we mean by converged results. We consider the total spectrum converged to x for a certain set of parameters if increasing any one of the parameters produces a relative discrepancy of at most x . The relative discrepancy of two spectra σ_1 and σ_2 is defined as

$$\epsilon(E) = 2 \frac{|\sigma_1(E) - \sigma_2(E)|}{\sigma_1(E) + \sigma_2(E)},$$

resp. by the maximum of this function. The maximum/minimum angles are considered converged to x if the absolute value of the difference upon increasing one parameter does not exceed x .

These definitions still leave room for interpretation as we only specify “increasing one parameter.” What we mean by that depends on the actual parameter and the behaviour upon changing it (see Appendices B.1 and B.2).

Furthermore, there are certain dependencies between convergence parameters. For example, increasing the complex scaling radius without changing the number of radial grid points corresponds to picking a smaller momentum scale; therefore, when investigating convergence with respect to this parameter, we will simultaneously increase the radial basis size and, as it turns out, also the maximal angular momentum.

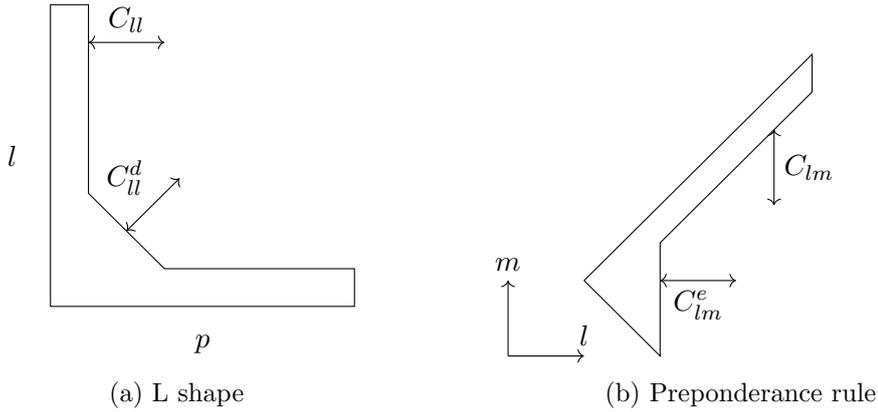


Figure 3.7.: The modified constraints used in practice

3.3.2. Convergence Parameters

The discretization is described by the following parameters, where we approach $L^2(\mathbb{R}^6)$ as each one tends towards infinity. The radial basis is given in FEM-DVR [49] with a total number of N grid points inside the complex scaling radius R_c and N_a grid points outside using `irECS`. The N grid points are distributed over n_{FE} finite elements (we do not consider n_{FE} a convergence parameter; however, we assume the radial basis to be too small if, for fixed R , the choice of n_{FE} impacts the result). Summing up, the convergence parameters for the radial basis are given by N , N_a and R_c .

The angular basis is defined by $l = 0, \dots, L - 1$, $|m| \leq l$ subject to the two earlier discussed constraints. In practice it turns out that one has to modify these slightly to allow for correlations for small l . The “L shape” constraint is defined by

$$(l < C_U) \vee (p < C_U) \vee (l + p < C_U^d),$$

that is, we allow for an additional triangular region $l + p < C_U^d$; see Fig. 3.7a. The preponderance rule $l - m < C_{lm}$ is modified to allow for

$$(l - m < C_{lm}) \vee l < C_{lm}^e,$$

that is, for small l we switch the constraint off; see Fig. 3.7b. Of course, the basis of the second electron is subject to the same constraint. Therefore the angular basis is defined by the convergence parameters L , C_{lm} , C_{lm}^d , C_U and C_U^d .

In addition to the convergence parameters describing the discretization, there are other numerical aspects to be controlled. First of all, the propagation uses RK4 with dynamic step size control; this mechanism requires to set an *accuracy*, denoted by ϵ . To compute `tSURFF` spectra, we average over the amplitude for times in a certain interval t_{Avg} after the pulse has stopped [51]. This is an important convergence parameter in XUV problems; for higher wavelengths we do not expect much influence

as the electron remaining in the ion sits in the ground state with a very high probability. Therefore, we do not expect much to happen after the pulse has finished (see Appendix B.1).

In addition to the complex scaling radius R_c , where \mathfrak{tSURFF} cuts off the potential, we introduce $R_s < R_c$, the smoothing radius, and cut off the potentials in the interval $[R_s, R_c]$ smoothly. Both these parameters correspond to a quite immediate alteration of the physics of the problem; therefore, we expect these parameters to be “interesting.”

Finally, as mentioned in Section 3.2.3, time propagation in realistic times is made feasible by employing a *spectral cut*. This means, we project out high energies of the Hamiltonian (the wave-function is smoothed out) to reduce stiffness of the problem. This yields a significant (in fact, without it none of the calculations presented could have been finished in time) speed-up. The energy above which all eigenvalues are removed is denoted by E_{cut} .

3.3.3. The Pulse

We use a 400nm pulse, polarized in the xy plane, as defined in Fig. 3.8. The envelope is given by a \cos^2 shape with full width at half maximum (FWHM) of two optical cycles. The pulse consists of two components, one along the x and one along the y axis, each of peak intensity $5 \times 10^{14} \text{W/cm}^2$, giving us a total intensity of $I = 10^{15} \text{W/cm}^2$. (The additivity for orthogonal components can be seen from the Poynting vector.)

The ponderomotive potential and Keldysh parameter of the pulse are given, using the ionization potential of the Helium ground state of approximately 0.903au, by

$$\begin{aligned} U_p &\approx 8.96 \text{eV}, \\ \gamma &\approx 1.17. \end{aligned}$$

Therefore, the situation lies in the cross-over regime between tunneling ($\gamma \ll 1$) and multi-photon ionization regimes ($\gamma \gg 1$) [2]. Furthermore, the number of photons required for ionization is given by

$$n_\gamma \approx \frac{0.903 \text{au}}{\hbar\omega} \approx \frac{24.6 \text{eV}}{3.10 \text{eV}} \approx 8.$$

The norm of the field strength reaches its maximum polarized along the positive y axis corresponding to an angle $\phi = \pi/2$. Therefore, in our crude approximation we expect the maximum ionization to appear along the positive x axis, or $\phi_{\text{max}} = 0$, deflected towards positive ϕ_{max} .

3.3.4. The Helium Model

As suggested in [4], the physics can be captured in a model where one electron moves and the other one remains in the ground state. We describe this situation using the Hamiltonian (on $L^2(\mathbb{R}^3)$)

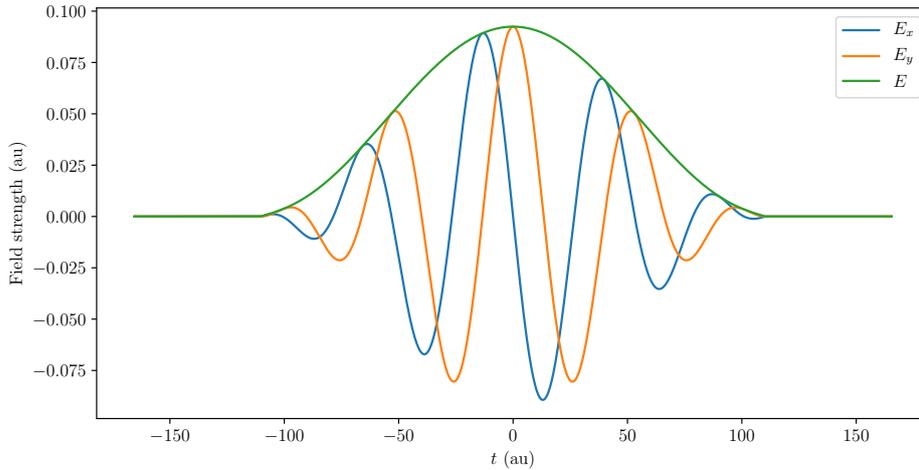


Figure 3.8.: The 400nm pulse used in this section. Given are the x and y components of the field strength, as well as the norm of it.

$$H = -\frac{\Delta}{2} - \frac{1}{r} - \frac{\exp[-\kappa r]}{r}, \quad (3.1)$$

i. e. the contributions of the second electron are represented by a shielded (Yukawa) potential. The parameter κ is chosen such that the ground state of this model matches the ionization potential of Helium (we obtain $\kappa \approx 2.135$).

This model enables us to do quick computations (times well below one hour) and investigate convergence with respect to certain parameters, which we do not expect to change in a full calculation, most notably the expansion size of the angular discretization, given by L , C_{lm} , and C_{lm}^e . The intuition behind this is that large L are required to model the parts of σ , where k is large. Big angular momentum is associated with big momentum; in these situations, where one electron moves outward with high kinetic energy, we do not expect two-electron correlations to have a big impact. Furthermore, the angular constraint C_{lm} and C_{lm}^e is essentially determined by the pulse polarization and envelope; therefore, we expect to get good agreement of these convergence parameters between the Helium model and full Helium.

The radial basis size used to obtain converged results in the model calculation can serve as a starting point for full Helium calculations; however, a priori one expects the full calculation to require a higher density of grid points. The reason for this is that motion in the Helium model essentially happens on Hydrogen scales as the shielded potential only distorts regions close to the origin. However, in the full Helium situation both electrons are in motion and are subject to the doubly ionized Helium scale of twice as high momenta.

The same reasoning applies to the spectral cut E_{cut} . This constraint smooths out the allowed wave-functions. The implications of this smoothing on the Coulomb

repulsion of both electrons are a priori unclear.

Figs. 3.9 and 3.10 give our results. We consider these converged with respect to all parameters *except for the tSURFF radius*; the next section gives further details on this. The total spectrum is converged to relative errors of 3%, whereas emission angles are considered converged to 1° , so we can expect to make predictions on the level of accuracy of 2° achieved in [1]. Convergence is shown in Appendix B.1.

The results show predominance of tunnel-ionization; however multi-photon peaks can be seen in the total spectrum as well as in the ϕ and k resolved one. From the latter, we see that multi-photon emission happens along the minimum of emission; more precisely, this process does not show a dependence with respect to ϕ , but along the minimum of tunnel-ionization, multi-photon ionization becomes discernible from tunnel ionization. We can identify a deflection angle of ten to twenty degrees; exact results are given in Section 3.3.6.

3.3.5. Potential Cutoff

Using the Volkov solutions in tSURFF amounts to an exact solution with the potential (all parts of the Hamiltonian except for kinetic energy and Laser field) set to zero outside of R_c ; we choose to make this explicit by cutting off the potential in the interval $[R_s, R_c]$. That is, outside R_c the Coulomb attraction exerted on the electron is neglected. As the Coulomb interaction is of long-range nature, this is a quite severe approximation. Fig. 3.11 and Tab. 3.1 show spectra and angles obtained with complex scaling radius of up to 49au. The observables cannot be considered converged in this range, where the cutoff appears to impact the total spectrum stronger than the maximum/minimum angles. As expected, the impact is most prominent for small momenta.

However, our goal is to investigate the impact electron-electron correlation has on the deflection angle. These correlations are expected to be stronger at smaller radii, where the electrons are closer separated and therefore the Coulomb repulsion, the force creating correlations between both electrons, is stronger. To estimate the impact on the deflection angle it is therefore sufficient to compare fully-correlated results with the uncorrelated single-electron model (3.1) for fixed $R_c = 21\text{au}$ and $R_s = 14\text{au}$.

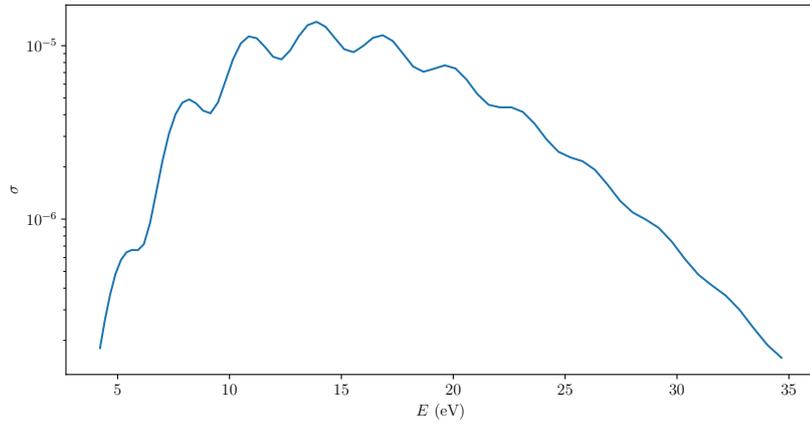
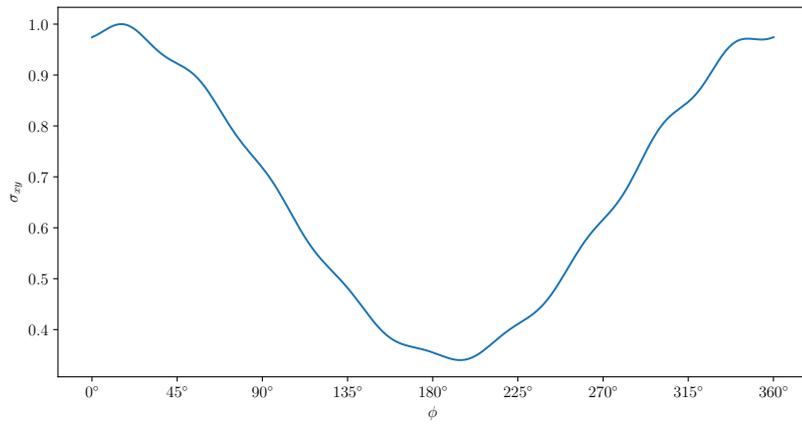
(a) Total spectrum σ (b) Spectrum resolved with respect to ϕ in the xy plane σ_{xy}

Figure 3.9.: Converged observables for the Helium model

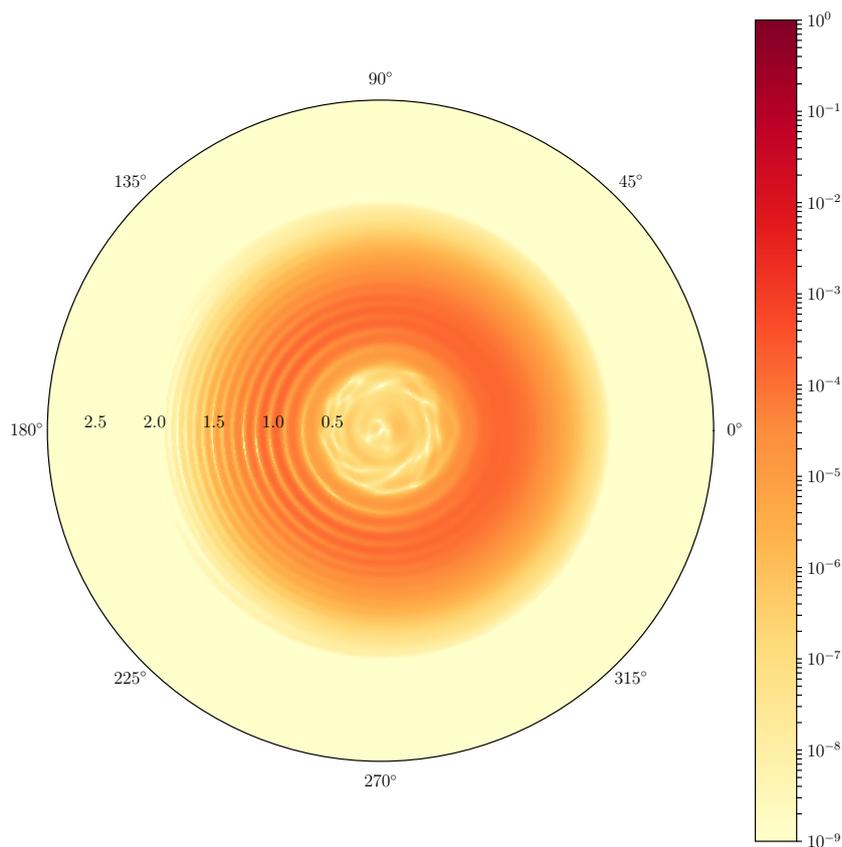


Figure 3.10.: Spectrum resolved with respect to k and ϕ for $\eta = 0$

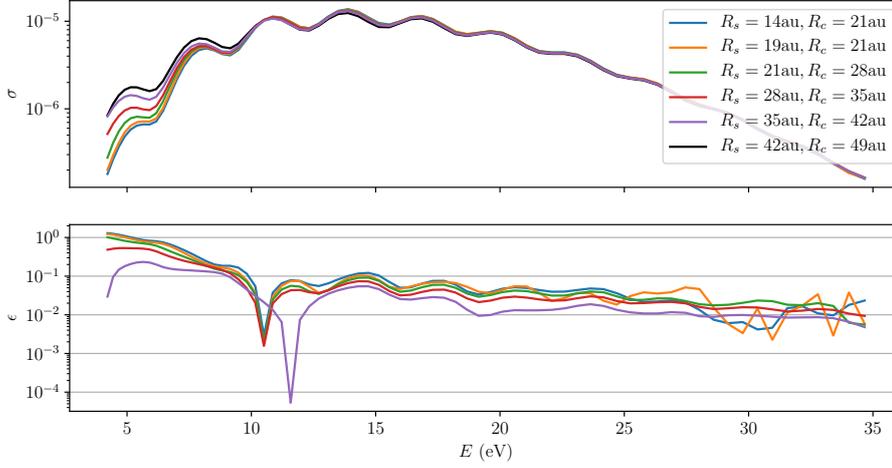


Figure 3.11.: Total spectra of the Helium model for various values of potential smoothing and cutoff radius. Convergence cannot be achieved in the range $R_c \leq 49\text{au}$.

	ϕ_{\max}	ϕ_{\min}	$\Delta\phi_{\max}$	$\Delta\phi_{\min}$
$R_s = 14\text{au}, R_c = 21\text{au}$	15.3°	193.9°	11.8°	15.5°
$R_s = 19\text{au}, R_c = 21\text{au}$	16.9°	194.3°	12.8°	17.6°
$R_s = 21\text{au}, R_c = 28\text{au}$	15.5°	192.8°	14.8°	20.2°
$R_s = 28\text{au}, R_c = 35\text{au}$	15.2°	192.5°	14.7°	19.7°
$R_s = 35\text{au}, R_c = 42\text{au}$	15.0°	192.4°	14.5°	19.7°
$R_s = 42\text{au}, R_c = 49\text{au}$	15.3°	192.7°	14.4°	19.4°

Table 3.1.: Emission angles of the Helium model for different potential cutoff and smoothing radii. Convergence cannot be established clearly within the range $R_c \leq 49\text{au}$.

3.3.6. Full Helium

The full Helium Hamiltonian on the Hilbert space $L^2(\mathbb{R}^6)$, where we set the nuclear mass to infinity, is given in atomic units by

$$H\psi(\vec{r}_1, \vec{r}_2) = \left(-\frac{1}{2}\Delta_{\vec{r}_1} - \frac{1}{2}\Delta_{\vec{r}_2} - \frac{2}{|\vec{r}_1|} - \frac{2}{|\vec{r}_2|} + \frac{1}{|r_1 - r_2|} \right) \psi(\vec{r}_1, \vec{r}_2).$$

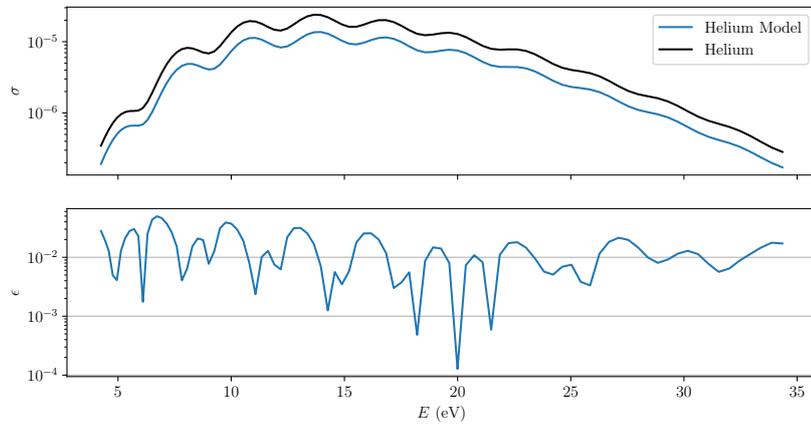
Figs. 3.12 and 3.13 and Tab. 3.2 present the converged results of our full Helium calculations. We consider the total spectrum converged to 5% and the angles converged to 1° . Convergence is discussed in Appendix B.2. Ionization in the Helium model happens with probability 1.5×10^{-4} , whereas the full Helium yields $2 \times 1.3 \times 10^{-4}$, about fifteen to twenty percent less than the model calculation. However, the shapes of the spectra agree to a few percent (see Fig. 3.12). We did not investigate this closer as we are mostly concerned with emission angles.

We see that the emission angles of Helium and the Helium model agree to about 2° . This appears to be a quite large difference compared to the convergence achieved. However, it is worth pointing out that the spectrum σ_{xy} is converged to 1%; from Fig. 3.12b we see that the discrepancy can be explained in terms of slightly different shapes. These probably originate from multi-photon effects, which are less strong in an IR calculation (see [4], where the maxima and minima can be determined much clearer).

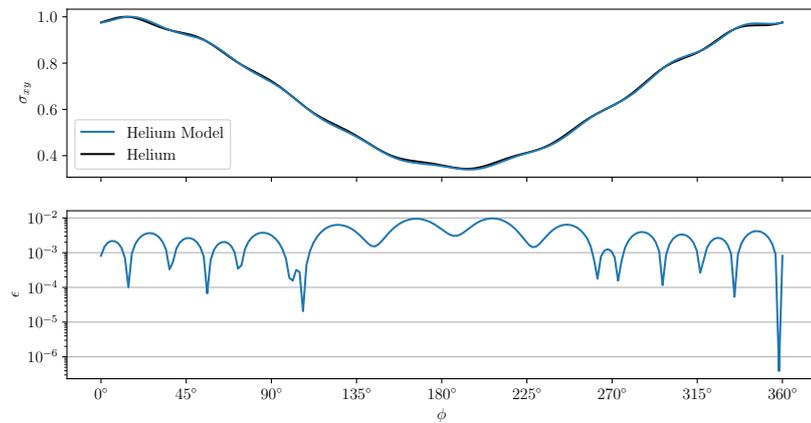
Therefore, we reason a discrepancy of about two 2° does not imply a strong influence of correlation effects, which is consistent with the fact that emission only happens in the ground state channels (see Fig. 3.13) as well as the insight that C_l^d has little impact on the angles (see Appendix B.2). The reasoning here is that two-electron correlations arise near the nucleus, where both angular momenta are small; if such processes contribute to the single-ionization spectra, one would expect the size of the angular discretization around $l = 0$ to play a major role. Furthermore, we note that the discrepancy is in the direction of smaller deflection angles, whereas the experiment measures significantly larger angles.

	ϕ_{\max}	ϕ_{\min}	$\Delta\phi_{\max}$	$\Delta\phi_{\min}$
Helium Model	15.3°	193.9°	11.8°	15.5°
Helium	13.1°	192.8°	11.6°	14.8°

Table 3.2.: Converged emission angles for the full Helium calculation compared to the Helium model



(a) Total spectrum σ . The comparison (lower plot) is achieved using spectra normalized to yield one.



(b) Spectrum resolved with respect to ϕ in the xy plane σ_{xy}

Figure 3.12.: Converged total spectrum and emission angles of the full Helium calculation compared to the Helium model

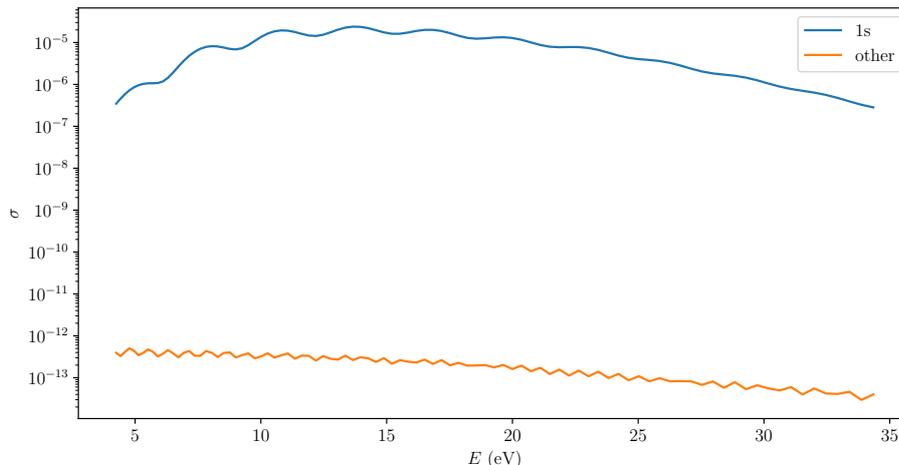


Figure 3.13.: Channel resolved total spectrum of the full Helium calculation. All emission happens in the ground state channel $1s$. The suppression of other channels to the level of numerical noise seems suspicious; however, calculations in the XUV regime using the exact same codebase have been made and qualitatively agree with [55].

3.4. Conclusion

The computed Helium and Helium model spectra agree within the accuracy to which we consider the spectra converged. Emission σ_{xy} in the xy plane also agrees well between the single active electron calculation and the fully-correlated one; we find maximum and minimum angles to differ by less than 2.5° . We consider the results converged to 1° , so this suggests an impact of electron-electron correlation. However, these angles are heavily influenced by the shape of σ_{xy} , an effect of the comparatively small wavelength, which makes for higher importance of multi-photon effect compared to an IR calculation [4]. Furthermore, the deflection angle in the fully-correlated calculation is smaller than in the model calculation, enlarging the discrepancy with respect to the experiment.

These results are not surprising as they are in good alignment with earlier results obtained using `haCC` [4]. Although we do not consider our calculations fully converged, the results are a strong hint that two-electron-correlations cannot explain excessive deflection angles measured in experiment [1]. To show this beyond doubt, one would need to conduct these calculations in the experimental setting, i. e. use an IR, elliptically polarized Laser. Such a calculation is feasible in the `tRecX` framework, owed to `tSURFF` and the use of angular constraints [4, 48].

Therefore, the discrepancy between experimental and theoretical (numerical) results continues to exist.

A. Some Calculations in Greater Detail

A.1. MPS Transfer Matrices are Completely Positive

Recall the transfer matrix $E : \mathbb{C}^{n \times n} \rightarrow \mathbb{C}^{m \times m}$, where we dropped the site index f and renamed the dimensions n and m :

$$E(\rho) = \sum_i A_i \rho A_i^\dagger$$

for matrices $A_i \in \mathbb{C}^{m \times n}$. To get acquainted with the terms positive and completely positive, we will now show that E is both.

Let $\rho \in \mathbb{C}^{n \times n}$ be positive, i. e. $\langle \psi, \rho \psi \rangle \geq 0$ for all $\psi \in \mathbb{C}^n$. Then

$$\begin{aligned} \langle \psi, E(\rho) \psi \rangle &= \sum_i \langle \psi, A_i \rho A_i^\dagger \psi \rangle \\ &= \sum_i \langle A_i^\dagger \psi, \rho A_i^\dagger \psi \rangle \geq 0. \end{aligned}$$

Therefore, $E(\rho)$ is positive and we call the map E positive.

Let now $k > 0$, and $\rho \in \mathbb{C}^{k \times k} \otimes \mathbb{C}^{n \times n} \cong \mathbb{C}^{kn \times kn}$ be a positive map. We need to show that $(\text{Id}_{k \times k} \otimes E)(\rho)$ is again positive in order to prove that E is completely positive. To this end we note that

$$(\text{Id}_{k \times k} \otimes E)(\rho) = \sum_i (\text{Id} \otimes A_i) \rho (\text{Id} \otimes A_i)^\dagger.$$

Therefore, $\text{Id}_{k \times k} \otimes E$ is of the same structure as E and we can apply the same proof of positivity. Let $\psi \in \mathbb{C}^{km} \cong \mathbb{C}^k \otimes \mathbb{C}^m$. Then

$$\begin{aligned} \langle \psi, (\text{Id}_{k \times k} \otimes E)(\rho) \psi \rangle &= \sum_i \langle \psi, (\text{Id} \otimes A_i) \rho (\text{Id} \otimes A_i)^\dagger \psi \rangle \\ &= \sum_i \langle (\text{Id} \otimes A_i)^\dagger \psi, \rho (\text{Id} \otimes A_i)^\dagger \psi \rangle \geq 0, \end{aligned}$$

which proves that E is completely positive.

A.2. Estimating Correlations in MPS

We start from (1.15), the correlation function of a right-canonical MPS (given in terms of transfer matrices $E_S^{A,1,B,2,C}$) with respect to the observable S

$$\begin{aligned}
\text{cov}_S(m, n) &= (E^A E_S^1 E^B E_S^2 E^C (1)) (E^A E^1 E^B E^2 E^C (1)) \\
&\quad - (E^A E_S^1 E^B E^2 E^C (1)) (E^A E^1 E^B E_S^2 E^C (1)) \\
&= \langle E^{A\dagger}(1), E_S^1 E^B E_S^2 (\text{Id}) \rangle \langle E^{A\dagger}(1), E^1 E^B E^2 (\text{Id}) \rangle \\
&\quad - \langle E^{A\dagger}(1), E_S^1 E^B E^2 (\text{Id}) \rangle \langle E^{A\dagger}(1), E^1 E^B E_S^2 (\text{Id}) \rangle \\
&= \langle \Lambda, E_S^1 E^B E_S^2 (\text{Id}) \rangle \langle \Lambda, E^1 E^B E^2 (\text{Id}) \rangle \\
&\quad - \langle \Lambda, E_S^1 E^B E^2 (\text{Id}) \rangle \langle \Lambda, E^1 E^B E_S^2 (\text{Id}) \rangle,
\end{aligned} \tag{A.1}$$

with the inner product $\langle A, B \rangle = \text{tr}[A^\dagger B]$; $\Lambda = \Lambda^{(m-1)}$ (see (1.13)) is a diagonal density matrix. Also a factor of one was added to the first summand. Recall the singular value decomposition of E^B

$$E^B(\rho) = \sum_{\alpha} \sigma_{\alpha} u_{\alpha} \langle v_{\alpha}, \rho \rangle,$$

with normed singular vectors u_{α} and v_{α} and $\alpha = 1, \dots, \min\{D_n^2, D_{m+1}^2\}$. Inserting this into (A.1), we obtain

$$\begin{aligned}
\text{cov}_S(m, n) &= \sum_{\alpha\beta} \sigma_{\alpha} \sigma_{\beta} \left(\langle \Lambda, E_S^1(u_{\alpha}) \rangle \langle v_{\alpha}, E_S^2(\text{Id}) \rangle \langle \Lambda, E^1(u_{\beta}) \rangle \langle v_{\beta}, E^2(\text{Id}) \rangle \right. \\
&\quad \left. - \langle \Lambda, E_S^1(u_{\alpha}) \rangle \langle v_{\alpha}, E^2(\text{Id}) \rangle \langle \Lambda, E^1(u_{\beta}) \rangle \langle v_{\beta}, E_S^2(\text{Id}) \rangle \right) \\
&= \sum_{\alpha\beta} \sigma_{\alpha} \sigma_{\beta} \langle \Lambda, E_S^1(u_{\alpha}) \rangle \langle \Lambda, E^1(u_{\beta}) \rangle \\
&\quad \left(\langle v_{\alpha}, E_S^2(\text{Id}) \rangle \langle v_{\beta}, E^2(\text{Id}) \rangle - \langle v_{\alpha}, E^2(\text{Id}) \rangle \langle v_{\beta}, E_S^2(\text{Id}) \rangle \right) \\
&= \sum_{\alpha\beta} \sigma_{\alpha} \sigma_{\beta} \langle \Lambda, E_S^1(u_{\alpha}) \rangle \langle \Lambda, E^1(u_{\beta}) \rangle \\
&\quad \left(\langle v_{\alpha}, E_S^2(\text{Id}) \rangle \text{tr}[v_{\beta}] - \langle v_{\beta}, E_S^2(\text{Id}) \rangle \text{tr}[v_{\alpha}] \right).
\end{aligned}$$

We recognize that terms with $\alpha = \beta$ do not contribute; specifically, $\sigma_1\sigma_1$ does not contribute. We will be able to estimate the correlation against the second largest singular value of E^B [39]. Therefore, we obtain

$$|\text{cov}_S(m, n)| \leq \sum_{\alpha \neq \beta} \sigma_\alpha \sigma_\beta \left| \langle E_S^{1\dagger}(\Lambda), u_\alpha \rangle \right| \left| \langle E_S^{1\dagger}(\Lambda), u_\beta \rangle \right| \\ \left(|\langle v_\alpha, E_S^2(\text{Id}) \rangle| |\text{tr}[v_\beta]| + |\langle v_\beta, E_S^2(\text{Id}) \rangle| |\text{tr}[v_\alpha]| \right).$$

We can bound the inner products using Cauchy-Schwartz, as the singular vectors are normed. Furthermore, denoting by $\|\circ\|$ the norm induced by $\langle \circ, \circ \rangle$, the Frobenius norm, we have

$$|\text{tr}[v_\alpha]| = |\langle \text{Id}, v_\alpha \rangle| \leq \|v_\alpha\| \|\text{Id}\| = \sqrt{\text{tr}[\text{Id}]} = \sqrt{D}.$$

Hence

$$|\text{cov}_S(m, n)| \leq 2\sqrt{D} \left\| E_S^{1\dagger}(\Lambda) \right\| \left\| E_S^{1\dagger}(\Lambda) \right\| \left\| E_S^2(\text{Id}) \right\| \sum_{\alpha \neq \beta} \sigma_\alpha \sigma_\beta.$$

Let us now bound the three vector norms in front of the sum. The second one is the most straight-forward as $E_S^{1\dagger}(\Lambda)$ is again a density matrix by (1.13). The Frobenius norm of a density matrix is bounded by one, where equality is attained for a pure state. For the other two vectors, we start by noting that the Frobenius norm can be bounded by the operator norm. Let $A \in \mathbb{C}^{D \times D}$ be a matrix with singular values $\lambda_1 \geq \lambda_2 \dots$. The operator norm is given by

$$\|A\|_{\text{op}} = \max_{\psi \in \mathbb{C}^D, \|\psi\|=1} \|A\psi\| = \lambda_1.$$

From the SVD, we see that the Frobenius norm of A is given by

$$\|A\| = \sqrt{\sum_i \lambda_i^2} \leq \sqrt{D} \lambda_1 = \sqrt{D} \|A\|_{\text{op}}.$$

To calculate the operator norm of $E_S^2(\text{Id})$ and $E_S^{1\dagger}(\Lambda)$, we note that we can always choose a basis of the factor space such that the observable S is diagonal. Therefore, we recall

$$E_S^2(\text{Id}) = \sum_i \lambda_i A_i A_i^\dagger,$$

where λ_i are the eigenvalues of S and we dropped the site-labelling index of A for readability. From the definition of right-canonical representation (1.13) we have

$$\sum_i A_i A_i^\dagger = \text{Id}.$$

As $E_S^2(\text{Id})$ is self-adjoint, the operator norm is given by $\max_{\|\psi\|=1} |\langle \psi, E_S^2(\text{Id})\psi \rangle|$ and, as we have

$$\begin{aligned} |\langle \psi, E_S^2(\text{Id})\psi \rangle| &= \left| \sum_i \lambda_i \langle A_i^\dagger \psi, A_i^\dagger \psi \rangle \right| \\ &\leq \max_i |\lambda_i| \sum_i \langle A_i^\dagger \psi, A_i^\dagger \psi \rangle \\ &= \|S\|_{\text{op}} \|\psi\|^2, \end{aligned}$$

the operator norm of $E_S^2(\text{Id})$ can be estimated against the operator norm of S . With a similar argument, we can bound the operator norm of $\|E_S^{1\dagger}(\Lambda)\|_{\text{op}} \leq \|S\|_{\text{op}}$. Therefore

$$|\text{cov}_S(m, n)| \leq 2D^{\frac{3}{2}} \|S\|_{\text{op}}^2 \sum_{\alpha \neq \beta} \sigma_\alpha \sigma_\beta.$$

For the final step, we estimate the largest singular value of E^B , or equivalently $E^{B\dagger}$, to arrive at an inequality in terms of σ_2/σ_1 . Pérez-García [56] proves that any trace preserving, positive map $T : \mathbb{C}^{n_1 \times n_1} \rightarrow \mathbb{C}^{n_2 \times n_2}$ satisfies $\|T\|_{\text{op}} \leq \sqrt{n_1}$ giving us $\sigma_1 \leq \sqrt{D}$. So, altogether we have

$$\begin{aligned} |\text{cov}_S(m, n)| &\leq 2D^{\frac{5}{2}} \|S\|_{\text{op}}^2 \sum_{\alpha \neq \beta} \frac{\sigma_\alpha \sigma_\beta}{\sigma_1^2} \\ &\leq 2D^{\frac{13}{2}} \|S\|_{\text{op}}^2 \frac{\sigma_2}{\sigma_1}, \end{aligned}$$

as the sums over α and β run over maximally D^2 values each. The power of D is quite unsatisfying in this result; however, our estimates are quite crude, better ones can be obtained (for example, by employing the nuclear norm of E^B to estimate the sum over β). Hastings [57] suggests that bounds independent of D can be found for sites located far enough from the endpoints. The important result here is that the correlation of sites n and m can be estimated against the second largest singular value of the transfer matrix between both sites.

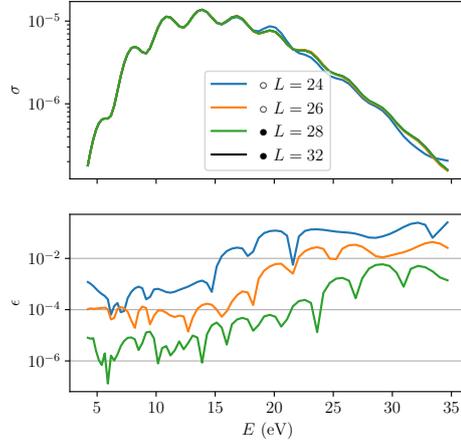
B. Convergence of Observables

B.1. Helium Model

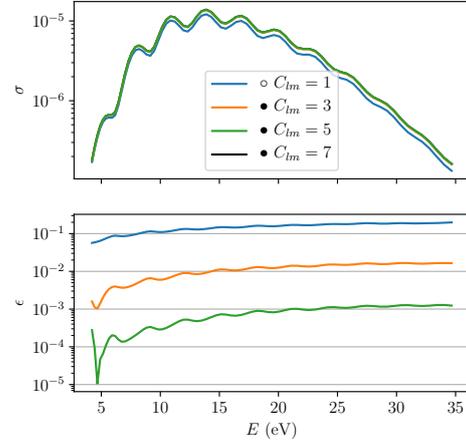
The convergence parameters are presented in Tab. B.1; plots to show convergence of total spectra are given in Figs. B.1 and B.2. Convergence of the angles is presented in Tabs. B.2 and B.3.

Total spectrum converged to	3%	5%
Angles converged to	1°	3°
L	28	26
C_{lm}	3	3
C_{lm}^e	6	4
N	48	36
ϵ	10^{-6}	10^{-6}
N_a	15	10
E_{cut}	50au	50au

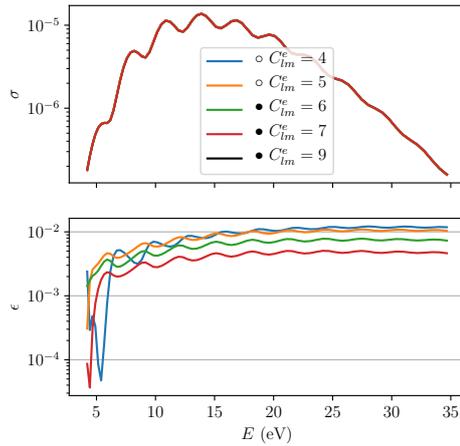
Table B.1.: Values for convergence of total spectrum to 3% resp. 5%, emission angles to 1° resp. 3° for the Helium model



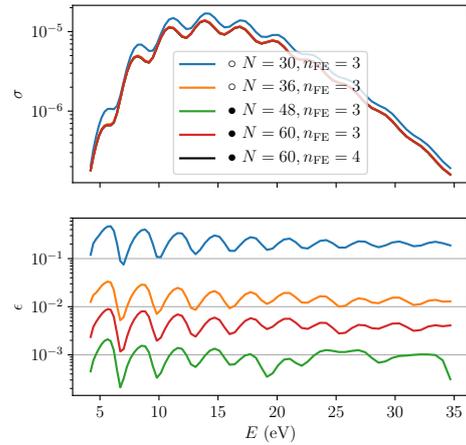
(a) Angular basis L



(b) Width of constraint C_{lm} , see Fig. 3.7b



(c) Exclusion width of constraint C_{lm}^e , see Fig. 3.7b

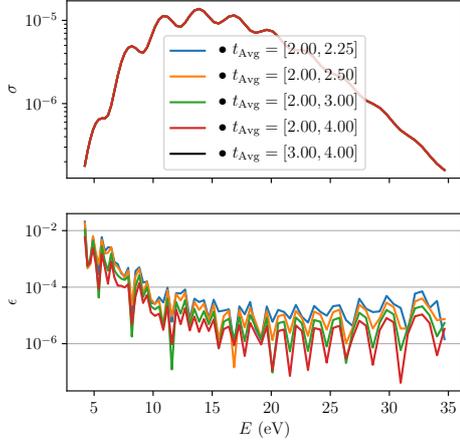


(d) Radial basis N

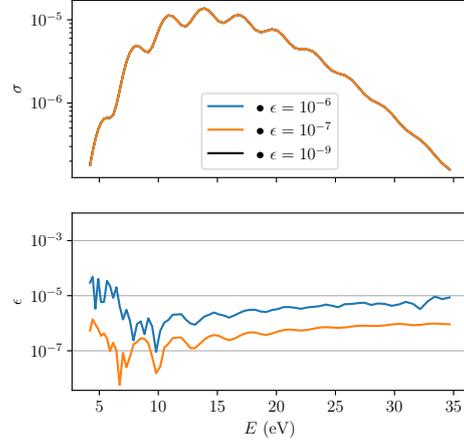
Figure B.1.: Convergence of the total spectrum of the Helium model with respect to radial and angular basis; filled-out circles represent converged results (total spectrum to 3%, emission angles to 1°).

	ϕ_{\max}	ϕ_{\min}	$\Delta\phi_{\max}$	$\Delta\phi_{\min}$
○ $L = 24$	15.5°	194.0°	11.7°	15.3°
○ $L = 26$	15.3°	193.9°	11.8°	15.5°
● $L = 28$	15.3°	193.9°	11.8°	15.5°
● $L = 32$	15.3°	193.9°	11.8°	15.5°
○ $C_{lm} = 1$	13.9°	193.7°	13.4°	15.7°
● $C_{lm} = 3$	15.3°	193.9°	11.8°	15.5°
● $C_{lm} = 5$	15.8°	194.1°	11.7°	15.5°
● $C_{lm} = 7$	15.7°	194.1°	12.3°	15.5°
○ $C_{lm}^e = 4$	12.5°	193.2°	11.9°	15.9°
○ $C_{lm}^e = 5$	13.9°	193.5°	11.8°	15.7°
● $C_{lm}^e = 6$	14.8°	194.0°	11.6°	15.4°
● $C_{lm}^e = 7$	15.3°	193.9°	11.8°	15.5°
● $C_{lm}^e = 9$	15.2°	193.7°	12.2°	15.9°
○ $N = 30, n_{\text{FE}} = 3$	19.1°	195.8°	13.1°	18.4°
○ $N = 36, n_{\text{FE}} = 3$	15.6°	194.1°	11.9°	15.6°
● $N = 48, n_{\text{FE}} = 3$	15.3°	193.9°	11.8°	15.5°
● $N = 60, n_{\text{FE}} = 3$	15.3°	193.9°	11.8°	15.5°
● $N = 60, n_{\text{FE}} = 4$	15.3°	193.9°	11.8°	15.5°

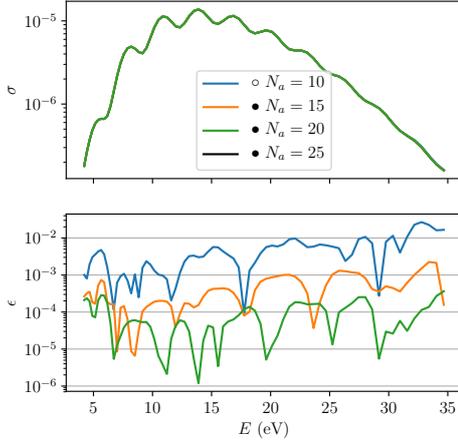
Table B.2.: Convergence of the emission angles of the Helium model with respect to radial and angular basis; filled-out circles represent converged results (total spectrum to 3%, emission angles to 1°).



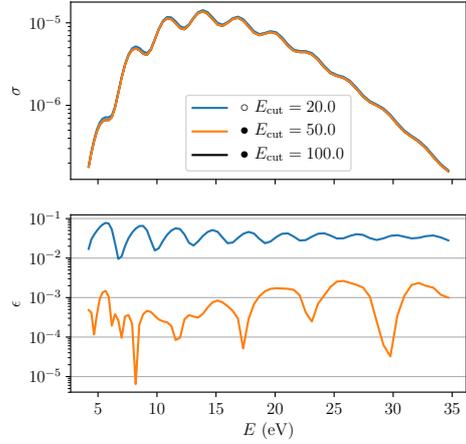
(a) Averaging interval t_{Avg} in units of optical cycles, see Section 3.3.2. As expected we see little impact of this convergence parameter.



(b) Accuracy of time propagation



(c) Radial basis outside of complex scaling radius N_a



(d) Spectral cut E_{cut}

Figure B.2.: Convergence of the total spectrum of the Helium model with respect to the remaining parameters; filled-out circles represent converged results (total spectrum to 3%, emission angles to 1°).

	ϕ_{\max}	ϕ_{\min}	$\Delta\phi_{\max}$	$\Delta\phi_{\min}$
• $t_{\text{Avg}} = [2.00, 2.25]$	15.3°	193.9°	11.8°	15.5°
• $t_{\text{Avg}} = [2.00, 2.50]$	15.3°	193.9°	11.8°	15.5°
• $t_{\text{Avg}} = [2.00, 3.00]$	15.3°	193.9°	11.8°	15.5°
• $t_{\text{Avg}} = [2.00, 4.00]$	15.3°	193.9°	11.8°	15.5°
• $t_{\text{Avg}} = [3.00, 4.00]$	15.3°	193.9°	11.8°	15.5°
• $\epsilon = 10^{-6}$	15.3°	193.9°	11.8°	15.5°
• $\epsilon = 10^{-7}$	15.3°	193.9°	11.8°	15.5°
• $\epsilon = 10^{-9}$	15.3°	193.9°	11.8°	15.5°
○ $N_a = 10$	15.3°	193.9°	11.8°	15.5°
• $N_a = 15$	15.3°	193.9°	11.8°	15.5°
• $N_a = 20$	15.3°	193.9°	11.8°	15.5°
• $N_a = 25$	15.3°	193.9°	11.8°	15.5°
○ $E_{\text{cut}} = 20.0$	15.5°	194.0°	12.0°	15.7°
• $E_{\text{cut}} = 50.0$	15.3°	193.9°	11.8°	15.5°
• $E_{\text{cut}} = 100.0$	15.3°	193.9°	11.8°	15.5°

Table B.3.: Convergence of the emission angles of the Helium model with respect to the remaining parameters; filled-out circles represent converged results (total spectrum to 3%, emission angles to 1°).

B.2. Helium

We start from the parameters set by the Helium Model (weaker convergence to 5% resp. 3°). E_{cut} is multiplied by two to arrive at the same cut energy on each factor space (see Section 3.2.3). We do not check convergence with respect to the parameters L , C_{lm} , ϵ and the averaging interval as we either estimate the impact very low (as is the case for the accuracy of time propagation and the averaging interval) or believe the parameters to be fixed by the model calculation. The parameter C_{lm}^e appears to impact emission angles quite strongly; therefore, we investigate convergence although we expect it to follow the same pattern as in the Helium model case.

To claim fully converged spectra, one would need to carry out the calculations with respect to the missing parameters and continue to push the total spectrum to a convergence below 1%. This requires a larger radial basis and wider C_{ll} ; although possible within the `tRecX` framework, such calculations would be much more time-consuming than the ones presented here.

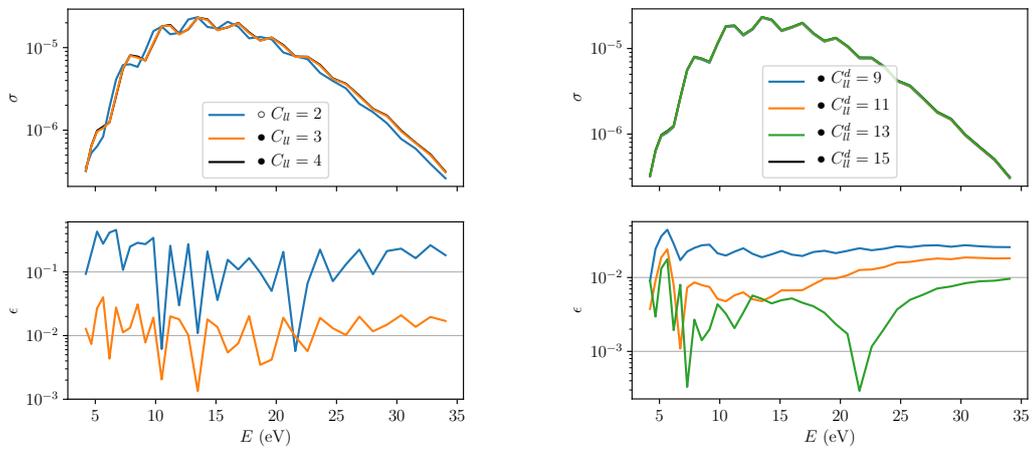
The remaining convergence parameters are given in Tab. B.4. Plots and angles are presented in Figs. B.3 and B.4 and Tab. B.5

Total spectrum converged to	5%
Angles converged to	1°
N	36
C_{lm}^e	6
C_{ll}	3
C_{ll}^d	9
E_{cut}	100au

Table B.4.: Values for convergence of total spectrum to 5%, emission angles to 1° for the full Helium calculation

	ϕ_{\max}	ϕ_{\min}	$\Delta\phi_{\max}$	$\Delta\phi_{\min}$
○ $C_{ll} = 2$	14.5°	197.4°	11.5°	16.1°
● $C_{ll} = 3$	10.4°	192.1°	11.8°	15.3°
● $C_{ll} = 4$	10.2°	191.8°	11.8°	15.4°
● $C_{ll}^d = 9$	10.4°	191.9°	11.7°	15.2°
● $C_{ll}^d = 11$	10.4°	192.1°	11.8°	15.3°
● $C_{ll}^d = 13$	10.5°	192.1°	11.8°	15.4°
● $C_{ll}^d = 15$	10.4°	192.0°	11.8°	15.5°
○ $C_{lm}^e = 4$	10.4°	192.1°	11.8°	15.3°
● $C_{lm}^e = 6$	12.6°	192.8°	11.4°	14.7°
● $C_{lm}^e = 7$	13.1°	192.8°	11.6°	14.8°
○ $N = 30, n_{\text{FE}} = 3$	13.3°	194.1°	12.2°	16.1°
● $N = 36, n_{\text{FE}} = 3$	10.5°	192.1°	11.8°	15.4°
● $N = 48, n_{\text{FE}} = 3$	10.8°	192.4°	11.9°	15.5°
● $N = 60, n_{\text{FE}} = 3$	10.9°	192.4°	11.9°	15.5°
● $E_{\text{cut}} = 100.0$	10.4°	192.1°	11.8°	15.3°
● $E_{\text{cut}} = 200.0$	10.4°	192.0°	11.8°	15.3°

Table B.5.: Convergence of the emission angles of Helium; filled-out circles represent converged results.



(a) Width of constraint C_U , see Fig. 3.7a

(b) Diagonal of exclusion C_U^d , see Fig. 3.7a

Figure B.3.: Convergence of the total spectrum of Helium, part one; filled-out circles represent converged results.

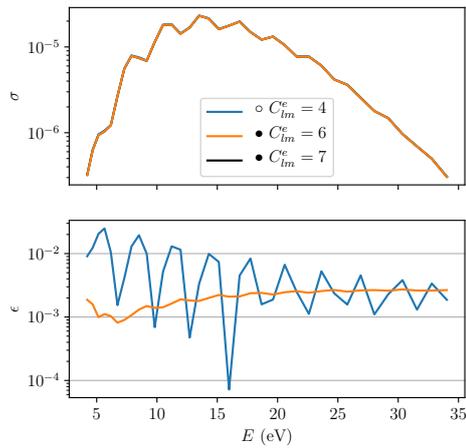
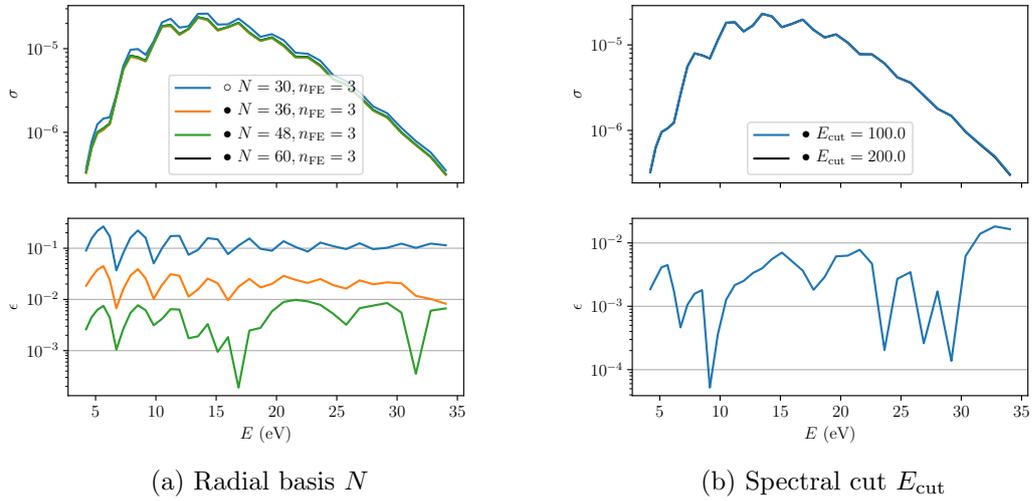


Figure B.4.: Convergence of the total spectrum of Helium, part two; filled-out circles represent converged results.

Bibliography

- [1] A. S. Landsman, M. Weger, J. Maurer, R. Boge, A. Ludwig, S. Heuser, C. Cirelli, L. Gallmann, and U. Keller, *Optica* **1**, 343 (2014).
- [2] A. S. Landsman, and U. Keller, *Phys. Rep.* **547**, 1 (2015).
- [3] L. Torlina, F. Morales, J. Kaushal, I. Ivanov, A. Kheifets, A. Zielinski, A. Scrinzi, H. G. Muller, S. Sukiasyan, M. Ivanov, and O. Smirnova, *Nat. Phys.* **11**, 503 (2015).
- [4] V. P. Majety, and A. Scrinzi, *J. Mod. Opt.* **64**, 1026 (2017).
- [5] M. H. Beck, A. Jäckle, G. A. Worth, and H.-D. Meyer, *Phys. Rep.* **324**, 1 (2000).
- [6] J. Zanghellini, M. Kitzler, C. Fabian, T. Brabec, and A. Scrinzi, *Laser Phys.* **13**, 1064 (2003).
- [7] I. Affleck, T. Kennedy, E. H. Lieb, and H. Tasaki, *Commun. Math. Phys.* **115**, 477 (1988).
- [8] M. Fannes, B. Nachtergaele, and R. F. Werner, *Commun. Math. Phys.* **144**, 443 (1992).
- [9] D. Pérez-Garcia, F. Verstraete, M. M. Wolf, and J. I. Cirac, *Quantum Inf. Comput.* **7**, 401 (2007).
- [10] C. Eckart, and G. Young, *Psychometrika* **1**, 211 (1936).
- [11] W. Hackbusch, *Computing* **62**, 89 (1999).
- [12] W. Hackbusch, and B. N. Khoromskij, *Computing* **64**, 21 (2000).
- [13] W. Hackbusch, and B. N. Khoromskij, *J. Comput. Appl. Math.* **125**, 479 (2000).
- [14] I. P. Gavriljuk, W. Hackbusch, and B. N. Khoromskij, *Numer. Math.* **92**, 83 (2002).
- [15] F. L. Hitchcock, *Journal of Mathematics and Physics* **6**, 164 (1927).
- [16] R. A. Harshman, *UCLA Working Papers in Phonetics* **16**, 1 (1970).
- [17] J. D. Carroll, and J.-J. Chang, *Psychometrika* **35**, 283 (1970).
- [18] T. G. Kolda, and B. W. Bader, *SIAM Rev.* **51**, 455 (2009).
- [19] J. B. Kruskal, *Linear Algebra Its Appl.* **18**, 95 (1977).
- [20] L. R. Tucker, *Psychometrika* **31**, 279 (1966).
- [21] L. D. Lathauwer, B. D. Moor, and J. Vandewalle, *SIAM J. Matrix Anal. Appl.* **21**, 1253 (2000).

- [22] L. R. Tucker, in *Problems in measuring change*, edited by C. W. Harris, University of Wisconsin Press, Madison WI, 1963, pp. 122–137.
- [23] A. Jäckle, and H.-D. Meyer, *J. Chem. Phys.* **104**, 7974 (1996).
- [24] D. Peláez, and H.-D. Meyer, *J. Chem. Phys.* **138**, 014108 (2013).
- [25] P. M. Kroonenberg, *Three-mode principal component analysis: theory and applications*, DSWO Press, Leiden, 1983.
- [26] L. D. Lathauwer, B. D. Moor, and J. Vandewalle, *SIAM J. Matrix Anal. Appl.* **21**, 1324 (2000).
- [27] L. Grasedyck, *SIAM J. Matrix Anal. Appl.* **31**, 2029 (2010).
- [28] P. A. M. Dirac, *Math. Proc. Camb. Philos. Soc.* **26**, 376 (1930).
- [29] J. Frenkel, *Wave mechanics*, Clarendon Press, Oxford, 1934.
- [30] K. Eckert, J. Schliemann, D. Bruß, and M. Lewenstein, *Ann. Phys. (N. Y.)* **299**, 88 (2002).
- [31] G. Beylkin, and M. J. Mohlenkamp, *Proc. Natl. Acad. Sci. U. S. A.* **99**, 10246 (2002).
- [32] O. Koch, W. Kreuzer, and A. Scrinzi, *Appl. Math. Comput.* **173**, 960 (2006).
- [33] J. Zanghellini, M. Kitzler, T. Brabec, and A. Scrinzi, *J. Phys. B* **37**, 763 (2004).
- [34] M. B. Hastings, *J. Stat. Mech. Theory Exp.* **2007**, P08024 (2007).
- [35] U. Schollwöck, *Ann. Phys. (N. Y.)* **326**, 96 (2011).
- [36] I. V. Oseledets, *SIAM J. Sci. Comput.* **33**, 2295 (2011).
- [37] G. Vidal, *Phys. Rev. Lett.* **91**, 147902 (2003).
- [38] D. M. Greenberger, M. A. Horne, and A. Zeilinger, in *Bell's theorem, quantum theory and conceptions of the universe*, edited by M. Kafatos, Kluwer Academic Publishers, Dordrecht, 1989, pp. 69–72.
- [39] R. Orús, *Ann. Phys. (N. Y.)* **349**, 117 (2014).
- [40] W. Bruzda, V. Cappellini, H.-J. Sommers, and K. Życzkowski, *Phys. Lett. A* **373**, 320 (2009).
- [41] M. B. Hastings, *Phys. Rev. B* **73**, 085115 (2006).
- [42] F. G. S. L. Brandão, and M. Horodecki, *Commun. Math. Phys.* **333**, 761 (2015).
- [43] R. Penrose, in *Combinatorial mathematics and its applications*, edited by D. J. A. Welsh, Academic Press, New York, 1971, pp. 221–244.
- [44] V. Murg, F. Verstraete, and J. I. Cirac, *Phys. Rev. A* **75**, 033605 (2007).
- [45] W. Hackbusch, and S. Kühn, *J. Fourier Anal. Appl.* **15**, 706 (2009).
- [46] Y.-Y. Shi, L.-M. Duan, and G. Vidal, *Phys. Rev. A* **74**, 022320 (2006).
- [47] G. Evenbly, *Gauge fixing, canonical forms and optimal truncations in tensor networks with closed loops*, 2018, [arXiv:1801.05390 \[quant-ph\]](https://arxiv.org/abs/1801.05390).

- [48] A. Zielinski, V. P. Majety, and A. Scrinzi, [Phys. Rev. A **93**, 023406 \(2016\)](#).
- [49] T. N. Rescigno, and C. W. McCurdy, [Phys. Rev. A **62**, 032706 \(2000\)](#).
- [50] L. V. Keldysh, *Sov. Phys. JETP* **20**, 1307 (1965).
- [51] L. Tao, and A. Scrinzi, [New J. Phys. **14**, 013021 \(2012\)](#).
- [52] A. Scrinzi, [New J. Phys. **14**, 085008 \(2012\)](#).
- [53] A. Scrinzi, [Phys. Rev. A **81**, 053845 \(2010\)](#).
- [54] B. Simon, [Phys. Lett. A **71**, 211 \(1979\)](#).
- [55] V. P. Majety, A. Zielinski, and A. Scrinzi, [New J. Phys. **17**, 063002 \(2015\)](#).
- [56] D. Pérez-Garcia, M. M. Wolf, D. Petz, and M. B. Ruskai, [J. Math. Phys. **47**, 083506 \(2006\)](#).
- [57] M. B. Hastings, *Notes on some questions in mathematical physics and quantum information*, 2014, [arXiv:1404.4327 \[quant-ph\]](#).