



TECHNISCHE  
UNIVERSITÄT  
WIEN

## DIPLOMARBEIT

# Machine Learning inspired Analysis of the Dyson Equation via Quantics Tensor Cross Interpolation

ausgeführt am Institut für Festkörperphysik  
der Technischen Universität Wien

unter der Anleitung von  
**Univ.Prof Dr.rer.nat. Karsten Held**  
**Dr.techn. Markus Wallerberger**

durch

**Alena Lorenz, BSc**  
Matrikelnummer: 01626524

April 29, 2024

\_\_\_\_\_  
Unterschrift StudentIn



Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar  
The approved original version of this thesis is available in print at TU Wien Bibliothek.

# Abstract

Confronting many-body problems in quantum field theory entails managing vast amounts of data, thus, being confronted with the challenge of balancing computational feasibility and accuracy of functional dependencies. This thesis elaborates on quantum tensor cross interpolation (QTCI), an innovative approach that merges two effective methods for handling tensors in multi-dimensional space-time: the quantics representation and tensor cross interpolation (TCI). QTCI benefits from both methods having distinct strategies in addressing numerical challenges. While the first method focuses on separating various length scales, TCI uses the restructured data to construct matrix product states (MPS) and compressing them, while maintaining an acceptable error. In this thesis, QTCI was used to solve the Dyson equation for the Hubbard model in one-, two-, and three-dimensional  $\mathbf{k}$ -space featuring a self-energy inspired by a self-energy deep in the Mott phase. It was found that the method is able to reliably and efficiently compress vast amounts of data while retaining an acceptable error. The computational effort increases with dimension, complexity of the function and accuracy which limits the applicability of QTCI depending on the computational resources available. Within the framework of this thesis, it was demonstrated that the maximum bond dimension  $D_{\max}$ , a measurement quantifying the complexity of functions compressed by QTCI, is linked to both the inverse temperature  $\beta$  and dimensionality  $n$  of the system by a universal power-law  $D_{\max}(\beta, n) = A_n \cdot \beta^{(n-1)0.253}$  with a dimension dependent factor  $A_n$ .



Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar  
The approved original version of this thesis is available in print at TU Wien Bibliothek.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>System and Method</b>	<b>3</b>
2.1	1-Particle Propagator . . . . .	3
2.2	Hubbard Model . . . . .	6
2.3	Matrix Product State (MPS) . . . . .	7
2.4	Skeleton Approximation . . . . .	11
2.5	Tensor Cross Interpolation (TCI) . . . . .	12
2.5.1	Finding pivots . . . . .	17
2.5.2	Quantics Tensor Cross Interpolation (QTCI) . . . . .	19
<b>3</b>	<b>Results and Discussion</b>	<b>24</b>
3.1	One dimension . . . . .	24
3.1.1	Error of QTCI . . . . .	30
3.1.2	Comparison of SVD and QTCI . . . . .	30
3.2	Two dimensions . . . . .	32
3.2.1	Error of QTCI . . . . .	34
3.2.2	Effect of tolerance $\epsilon$ on maximum bond dimension $D_{\max}$ . . . . .	35
3.2.3	Effect of chemical potential $\mu$ . . . . .	36
3.2.4	Fused Quantics Representation . . . . .	38
3.3	Three dimensions . . . . .	39
3.3.1	Fused Quantics Representation . . . . .	40
3.4	Universal Power Law for $D_{\max}$ . . . . .	41
<b>4</b>	<b>Conclusion and Outlook</b>	<b>43</b>
	<b>Bibliography</b>	<b>i</b>



Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar  
The approved original version of this thesis is available in print at TU Wien Bibliothek.

# 1 Introduction

Solid-state physics naturally deals with a large number of interacting particles of the scale of  $10^{24}$  making it impossible to keep track of every particle interaction. For most materials it is sufficient to combine interactions into an effective potential [1]. These materials are weakly correlated systems meaning that individual particle-particle interactions can be omitted and instead the collective interaction of all particles is used when calculating the motion of a single particle. Theories describing effective one-particle systems include density functional theory and the Hartree-Fock theory, two methods commonly used to determine electronic properties in material science. These theories yield accurate results when, for example, the interaction energy is significantly smaller than the kinetic energy or if the interaction length of the particle is by several scales smaller than the distance between interacting particles, leading to a screened potential and a weak effective interaction.

However, for some materials, like Mott insulators or superconductors, the correlation of particles is substantial for the material's properties, making it much harder to computationally deal with. The challenge is that correlated many-body systems inherently deal with countless degrees of freedom in an exponentially growing Hilbert space making it very expensive in both time and memory to reasonably work with them. The Green's function formalism of quantum field theory (QFT) gives us the tools to calculate the propagators of the particle under investigation and, thus, approximate the system's properties [2]. Other methods in QFT tackling correlated many-body problems include the Bethe-Salpeter equation which, in contrast to the one-particle Green's function, is used to evaluate the two-particle correlation function [3].

However, these functions are complex tensors in high-dimensional space-time becoming especially challenging to calculate with when moving to two- or three-dimensional space. Recent advances in data compression, notably through singular

value decomposition (SVD), have made the management of such data more feasible [4]. The  $n$ -dimensional tensor is transformed into its matrix product state (MPS) where each physical variable is assigned a set of singular values. By omitting the smallest singular values, the tensor gets condensed to its essential characteristics, thus, reducing its degrees of freedom. When using SVD the complete tensor must be available, thus, requiring sufficient memory.

Recently in 2023, an alternative approach was proposed [5] to combine two mathematical methods dealing with high-dimensional tensors, namely the quantum representation [6] and the skeleton approximation [7], creating quantum tensor cross interpolation (QTCI). Each method addresses the problem from a different angle, making QTCI a highly efficient combination of two promising techniques. Conceptually, the quantum representation rearranges the data, separating different length scales, a concept commonly used in renormalisation group theory [8]. On the other hand, building the MPS using the skeleton approximation allows one to effectively compress areas of less entangled length scales while keeping a high degree of accuracy. Since QTCI has been introduced rather recently the method is in its early phase of development and testing. This thesis aims at enhancing the understanding of the method by applying it to simple problems of quantum field theory. Three main objectives will be investigated, namely the compressibility of the tensor with varying parameters, like temperature or the chemical potential, the accuracy of the reconstructed function, and lastly the correlation between temperature and compressibility. Furthermore, comparing the performance of QTCI and SVD will give valuable insight on what method is when preferable over the other.

## 2 System and Method

The foundation of this thesis is formed by the Green's function formalism of QFT which will be briefly introduced in what follows. The next part will give a detailed account of the method used in this thesis, namely QTCI, and its mathematical motivation.

### 2.1 1-Particle Propagator

Studying solids by solving the Hamiltonian for huge numbers of particles is cumbersome as quantum systems are defined in the Hilbert space growing exponentially with the number of particles. Instead, one may be solely interested in the propagation of a single particle in the system corresponding to the probability of a particle at  $\mathbf{r}, t$  to be found at position  $\mathbf{r}'$  later in time  $t'$ . Experimentally, this is done by angle-resolved photoemission spectroscopy to calculate the single-particle spectral function. To experimentally determine the two-particle correlation function, resonant inelastic X-ray scattering [9] or neutron scattering are employed [10].

The path the particle takes in between is visualised by Feynman diagrams, see fig. 2.1, and the calculations are provided by the Green's function formalism.

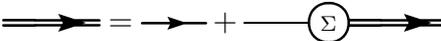
$$G = G_0 + G_0 \Sigma G$$


Figure 2.1: The interacting Green's function on the left is built from the unperturbed Green's function  $G_0$  (denoted as a single line) and the self-energy  $\Sigma$  containing all one-particle irreducible Feynman diagrams (and thus the interaction between particles).

As a first step, the Green's function [2, 11, 12] for one particle in the Heisenberg picture is introduced

$$G(\mathbf{r}', t', r, t) = -i\langle \hat{\psi}(\mathbf{r}', t'), \hat{\psi}^\dagger(\mathbf{r}, t) \rangle \theta(t' - t) \pm i\langle \hat{\psi}^\dagger(\mathbf{r}, t), \hat{\psi}(\mathbf{r}', t') \rangle \theta(t - t') \quad (2.1)$$

where the  $\pm$  refers to fermions (anti-symmetric) and bosons (symmetric) respectively. Causality is preserved by the Heaviside function meaning that a particle is added to (removed from) the system at  $t < t'$  ( $t' < t$ ) and removed (added) at some later point in time  $t'$  ( $t$ ). Reversing  $t$  and  $t'$  also changes the time ordering which can be reproduced by the time ordering operator  $\mathcal{T}$ . Performing a Wick's rotation  $t \leftrightarrow -i\tau \mid 0 < \tau < \beta$  transforms equation 2.1 to

$$G(\mathbf{r}', \tau', \mathbf{r}, \tau) = -\langle \mathcal{T} \hat{\psi}(\mathbf{r}', -i\tau') \hat{\psi}^\dagger(\mathbf{r}, -i\tau) \rangle. \quad (2.2)$$

As a next step, the definition of the thermal expectation value  $\langle \dots \rangle$  and the density matrix for the grand canonical distribution with the grand potential  $\Omega$  in the Schrödinger picture is used and the time ordering operator is omitted by setting  $\tau > \tau'$ ,

$$G(\mathbf{r}', \tau', \mathbf{r}, \tau) = -\mathbf{tr} \left( e^{\beta(\Omega - \mathcal{H})} e^{\mathcal{H}\tau} \psi(\mathbf{r}, \tau) e^{-\mathcal{H}\tau} e^{\mathcal{H}\tau'} \bar{\psi}(\mathbf{r}', \tau') e^{-\mathcal{H}\tau'} \right) \quad (2.3)$$

$$= -\frac{1}{Z} \mathbf{tr} \left( e^{-(\beta - \tau + \tau')\mathcal{H}} \psi(\mathbf{r}) e^{-\mathcal{H}(\tau - \tau')} \bar{\psi}(\mathbf{r}') \right). \quad (2.4)$$

Here the partition function  $Z = e^{-\beta\Omega}$  for the grand canonical ensemble is introduced. Cyclic invariance of the trace allows changing the order of the arguments in the second line. From equation 2.3 it is deduced that the propagator depends solely on the time difference  $\tau - \tau'$  ( $\tau' - \tau$ ) for  $\tau > \tau'$  ( $\tau' > \tau$ ) due to the time-independence of the Hamiltonian  $\mathcal{H}$ . Due to the (anti-)periodicity, the Fourier series expansion can be written as [13]

$$G(\mathbf{r}', 0, \mathbf{r}, \tau) = \frac{1}{\beta} \sum_{n=-\infty}^{\infty} G(\mathbf{r}, \mathbf{r}', \omega_n) e^{-i\omega_n \tau} \quad (2.5)$$

with the discrete Matsubara frequencies

$$i\omega_n = \frac{(2n+1)i\pi}{\beta} \quad \text{and} \quad i\omega_n = \frac{i\pi n}{\beta} \quad (2.6)$$

for fermions and bosons respectively. These frequencies are poles of the Fermi- (Bose-) function.

Calculations of an explicit expression depending on  $\omega_n$  for the unperturbed Green's function for fermions starts with equation 2.1

$$G_0(\mathbf{r}, t) = -i \langle \mathcal{T} \hat{\psi}(\mathbf{r}, t) \hat{\psi}^\dagger(0, 0) \rangle_0. \quad (2.7)$$

Rewriting the field operators in the basis of plane waves  $\hat{\psi}(\mathbf{r}, t) = \frac{1}{\sqrt{V}} \sum_{\mathbf{k}} a_{\mathbf{k}} e^{i\mathbf{k}\mathbf{r} - i(\epsilon_{\mathbf{k}} - \mu)t}$  changes the equation to

$$G_0(\mathbf{r}, t) = \frac{-i}{V} \sum_{\mathbf{k}, \mathbf{k}'} \langle \theta(t) a_{\mathbf{k}} a_{\mathbf{k}'}^\dagger - \theta(-t) a_{\mathbf{k}'}^\dagger a_{\mathbf{k}} \rangle e^{i\mathbf{k}\mathbf{r} - i(\epsilon_{\mathbf{k}} - \mu)t}. \quad (2.8)$$

Furthermore, the equilibrium average at zero temperature ( $T = 0$ )  $\langle a_{\mathbf{k}}^\dagger a_{\mathbf{k}'} \rangle_0 = \delta_{\mathbf{k}\mathbf{k}'} \theta(\mu - \epsilon_{\mathbf{k}})$  is considered

$$G_0(\mathbf{r}, t) = \frac{-i}{V} \sum_{\mathbf{k}, \mathbf{k}'} (\theta(t) (1 - \theta(\mu - \epsilon_{\mathbf{k}})) - \theta(-t) \theta(\mu - \epsilon_{\mathbf{k}})) e^{i\mathbf{k}\mathbf{r} - i(\epsilon_{\mathbf{k}} - \mu)t}. \quad (2.9)$$

As a last step, the Fourier transform is used to give a concise expression for the unperturbed Dyson equation

$$G_0(\mathbf{k}, \omega_n) = \frac{1}{i\omega_n - \epsilon_{\mathbf{k}} + \mu}. \quad (2.10)$$

For an interacting system,  $G_0$  is not sufficient to describe the system and the self-energy  $\Sigma$  is introduced, leading to an expression for the perturbed Green's function

$$G(\mathbf{k}, \omega_n) = (G_0(\mathbf{k}, \omega_n) - \Sigma(\mathbf{k}, \omega_n))^{-1} = \frac{1}{i\omega_n - \epsilon_{\mathbf{k}} + \mu - \Sigma(\mathbf{k}, \omega_n)}. \quad (2.11)$$

In terms of Feynman diagrams,  $\Sigma$  contains all one-particle irreducible diagrams.

## 2.2 Hubbard Model

The Hubbard model is used to study a handful of material exhibiting correlated electron systems in a lattice structure. It is an extension of the tight-binding model in the sense that it includes a pair-wise interaction term while keeping the tight-binding aspect of electrons being localised at the atom sites. Solids well-approximated by the Hubbard model may exhibit characteristics of both the nearly free electron model and the atomic limit, depending on the interaction that can be estimated by self-consistent methods like constrained random phase approximation (cRPA) [13]. The expression of the Hamiltonian in second quantisation is

$$\mathcal{H} = - \sum_{\langle i,j \rangle} \sum_{\sigma} t c_{i\sigma}^{\dagger} c_{j\sigma} - \mu \sum_{i,\sigma} n_{i\sigma} + U \sum_i n_{i\uparrow} n_{i\downarrow}. \quad (2.12)$$

The creation  $c_{i\sigma}^{\dagger}$  and annihilation  $c_{i\sigma}$  operators create or destroy the particle with spin  $\sigma \in \{\uparrow, \downarrow\}$  occupying the quantum state  $i$ . The number operator  $n_{i\sigma} = c_{i\sigma}^{\dagger} c_{i\sigma}$  counts the number of electrons at site  $i$  with spin  $\sigma$ . The first term corresponds to the *hopping* term which is defined by the *hopping* amplitude  $t$  allowing the electron to tunnel between sites, with the sum running over all nearest neighbours  $\langle i, j \rangle$ . The chemical potential  $\mu$  is proportional to the change in energy for adding an electron to the system. The final term characterises the on-site interaction between two electrons. Thereby,  $U$  is the interaction strength mimicking the screened Coulomb repulsion.

Tuning just a few parameters the model is able to describe a wide range of scenarios. For example, with  $U \ll t$  the on-site electron interaction becomes small, resembling a weakly interacting electron system. In contrast,  $U \gg t$  at half-filling allows for little hopping leading to an insulating behaviour. The Hubbard model upholds the Pauli exclusion principle as well as the localisation of the electrons by discrete sums over the atom sites.

Solving the Schrödinger equation for half-filling and  $U = 0$  in one dimension [1, 2], starts with the Fourier-transformed annihilation and creation operators

$$c_{i\sigma}^{\dagger} = \frac{1}{\sqrt{N}} \sum_k e^{ikr_i} c_{k\sigma}^{\dagger} \quad \text{and} \quad c_{i\sigma} = \frac{1}{\sqrt{N}} \sum_k e^{ikr_i} c_{k\sigma}. \quad (2.13)$$

Substitution into 2.12 yields:

$$\mathcal{H} = -\frac{1}{N} \sum_{\langle i,j \rangle} \sum_{\sigma} t \left( \sum_{\mathbf{k}} e^{i\mathbf{k}r_j} c_{\mathbf{k}\sigma}^{\dagger} \right) \left( \sum_{\mathbf{k}'} e^{-i\mathbf{k}'r_i} c_{\mathbf{k}'\sigma} \right) - \frac{\mu}{N} \sum_i \sum_{\mathbf{k},\sigma} c_{\mathbf{k}\sigma}^{\dagger} c_{\mathbf{k}\sigma} \quad (2.14)$$

$$= -\frac{1}{N} \sum_{\mathbf{k},\mathbf{k}'} \sum_{\sigma} t \left( e^{i\mathbf{k}'} + e^{-i\mathbf{k}'} \right) c_{\mathbf{k}\sigma}^{\dagger} c_{\mathbf{k}'\sigma} \sum_j e^{i\mathbf{r}_j(\mathbf{k}-\mathbf{k}')} - \mu \sum_{\mathbf{k},\sigma} c_{\mathbf{k}\sigma}^{\dagger} c_{\mathbf{k}\sigma} \quad (2.15)$$

In the last step, the relation  $r_i = r_j \pm a$  was used with  $a = 1$  being the lattice constant as only hopping between nearest neighbours is allowed.

$$\mathcal{H} = -\frac{1}{N} \sum_{\mathbf{k},\mathbf{k}'} \sum_{\sigma} t \left( e^{i\mathbf{k}'} + e^{-i\mathbf{k}'} \right) c_{\mathbf{k}\sigma}^{\dagger} c_{\mathbf{k}\sigma} N \delta(\mathbf{k} - \mathbf{k}') - \mu \sum_{\mathbf{k},\sigma} c_{\mathbf{k}\sigma}^{\dagger} c_{\mathbf{k}\sigma} \quad (2.16)$$

$$= -\sum_{\mathbf{k},\sigma} 2t \cos(\mathbf{k}a) c_{\mathbf{k}\sigma}^{\dagger} c_{\mathbf{k}\sigma} - \mu \sum_{\mathbf{k},\sigma} c_{\mathbf{k}\sigma}^{\dagger} c_{\mathbf{k}\sigma} \quad (2.17)$$

From the last line the eigenvalue can be easily deduced as

$$\epsilon_{\mathbf{k}} = -2t \cos \mathbf{k} - \mu. \quad (2.18)$$

Generalised to arbitrary dimension this equation gives

$$\epsilon_{\mathbf{k}} = -2 \sum_i t_i \cos k_i - \mu. \quad (2.19)$$

with the sum over  $i$  depending on the system's dimensions.

To determine the Fermi surface for an arbitrary dimension  $d$  the energy dispersion is plotted in the first Brillouin zone  $\mathbf{k} \in [-\pi, \pi]^d$ , see fig. 2.2a for  $d = 2$ . When holding the energy constant, lines form in  $k$ -space representing the Fermi surface up to which the electrons occupy energy states, see fig. 2.2b.

## 2.3 Matrix Product State (MPS)

Matrix Product States (MPS) are motivated by the complexity of quantum systems that grow exponentially with the degrees of freedom, an obstacle also known as "curse of dimensionality"[6, 14]. This can be demonstrated with a particle in three-dimensional space having three spatial degrees of freedom.

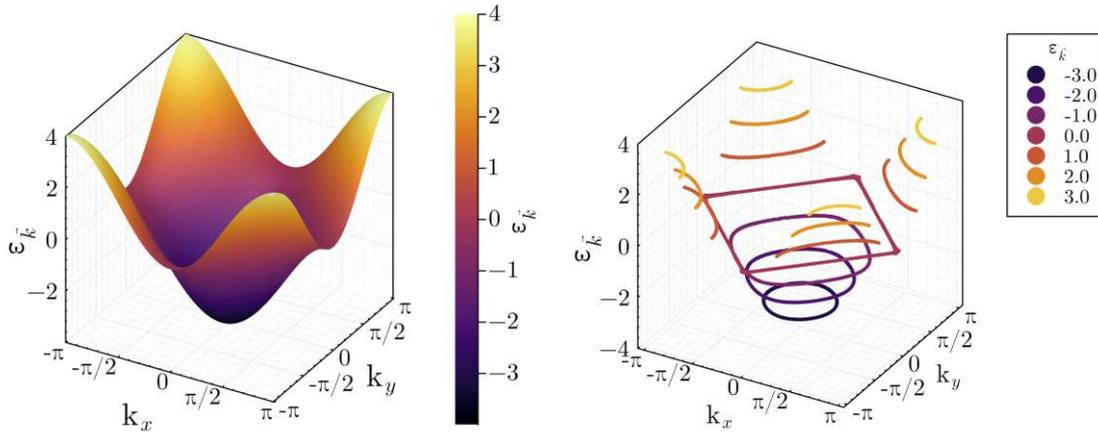


Figure 2.2: (a) energy dispersion in two dimensions. (b) depiction of the Fermi surface (lines of constant energy) for different discrete Fermi energies.

Thus, the particle is correctly describable in  $\mathbb{R}^3$ . When considering the particle's momentum, three more degrees of freedom have to be considered. Now a Hilbert space of six dimensions is required and for every particle added to the system six more degrees of freedom are added. This makes high-dimensional systems hard to study and one is quickly limited by the available computation time and memory.

The idea of MPS is to use compression in order to capture essential information omitting insignificant or repetitive data while retaining an acceptable accuracy, thereby reducing the computational cost and memory required. Mathematically, this means decomposing any state into a network of lower-order tensors and contracting over inner indices following a specific pattern, thereby exploiting less than maximum entanglement between the different degrees of freedom [14]. This is visualised in fig. 2.3 where an arbitrary matrix  $A$  with  $n$  physical indices  $u_1, u_2, \dots, u_n$  is gradually decomposed into its MPS form. Conceptually, the  $n$ -dimensional matrix transforms into the product of  $n$  matrices each taking one physical index  $u_i$  as argument. For the matrix multiplication sums over hidden indices (green lines in fig. 2.3) are introduced providing the possibility to manipulate the individual products. For a large enough number of hidden (green) indices, the MPS is an exact representation of  $A$  merely being rewritten into a different form. The advantage of MPS reveals

itself when introducing a cut-off, limiting the accuracy but simultaneously reducing memory and computation time. This is demonstrated when considering the singular value decomposition (SVD) for  $A$  [4]. SVD factorised any matrix into three matrices: a rotation  $V^\dagger$ , a rescaling  $S$  and another rotation  $U$ :

$$A = USV^\dagger. \quad (2.20)$$

The rectangular matrix  $S$  has only diagonal non-zero elements. These values are called the singular values  $S_{ii} = s_i$  sorted according to value,  $s_1$  being the largest of them. This representation of  $A$  can be truncated by setting all singular values smaller than a predefined threshold to zero. SVD can also be generalised to a matrix of dimension  $d^n$ , by grouping the indices accordingly.

In the case of MPS each index  $u_i$  one SVD can be performed, starting with  $u_1$  in this example (see the first line of fig. 2.3)

$$A(u_1, u_2, \dots, u_n) = A((u_1), (u_2, \dots, u_n)) = \sum_{\alpha_1}^{r_1} U_{u_1, \alpha_1} S_{\alpha_1, \alpha_1} (V^\dagger)_{\alpha_1, (u_2, \dots, u_n)} \quad (2.21)$$

$$= \sum_{\alpha_1}^{r_1} M_{1, \alpha_1}^{u_1} A_{(\alpha_1, u_2), (u_3, \dots, u_n)}. \quad (2.22)$$

controlling the accuracy of the new representation by truncating the singular values  $S_{\alpha_1, \alpha_1}$  at a suitable threshold  $r_1$ . For the second variable  $u_2$ , the truncated matrix  $S$  and  $V^\dagger$  were multiplied and the matrix  $U_{u_1, \alpha_1}$  was replaced by a set of  $d$  vectors  $M_{1, \alpha_1}^{u_1}$  before performing another SVD giving (second line of fig. 2.3)

$$A(u_1, u_2, \dots, u_n) = \sum_{\alpha_1}^{r_1} \sum_{\alpha_2}^{r_2} M_{1, \alpha_1}^{u_1} U_{(\alpha_1, u_2), \alpha_2} S_{\alpha_2, \alpha_2} (V^\dagger)_{\alpha_2, (u_3, \dots, u_n)} \quad (2.23)$$

$$= \sum_{\alpha_1}^{r_1} \sum_{\alpha_2}^{r_2} M_{1, \alpha_1}^{u_1} M_{\alpha_1, \alpha_2}^{u_2} S_{\alpha_2, \alpha_2} (V^\dagger)_{\alpha_2, (u_3, \dots, u_n)}. \quad (2.24)$$

$M_{\alpha_1, \alpha_2}^{u_2}$  can be interpreted as a set of  $d$  matrices. Repeating the SVD for each

index results in a truncated MPS

$$A(u_1, u_2, \dots, u_n) = \sum_{\alpha_1, \dots, \alpha_n} M_{1, \alpha_1}^{u_1} M_{\alpha_1, \alpha_2}^{u_2} \dots M_{\alpha_{n-1}, \alpha_n}^{u_{n-1}} M_{\alpha_{n-1}, 1}^{u_n} \quad (2.25)$$

as depicted in the last line of fig. 2.3. The sums over  $\alpha_i$  are indicated by the green lines. The higher the entanglement between the variables, the more terms have to be added up to attain the required accuracy.

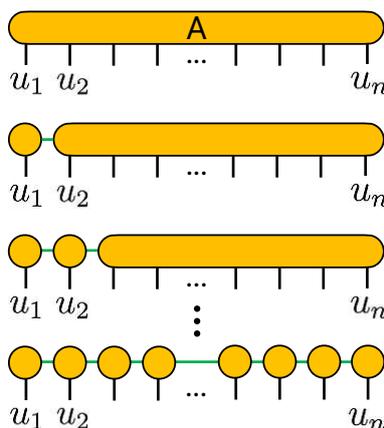


Figure 2.3: Conceptual construction of a MPS. Site by site the multi-dimensional tensor is deconstructed (from top to bottom) into the product of tensors of lesser dimension each taking one physical index as argument (black lines). The accuracy can be tuned by truncation of the sums over the hidden indices (green lines).

The SVD can be performed in any arbitrary order of  $u_i$  meaning that performing SVD for index  $u_2$  before  $u_1$  does not change the outcome.

Among the methods for obtaining a MPS besides SVD are [4], the Schmidt Decomposition [4] and the Tensor Cross Interpolation (TCI) [15, 16]. They all share the principle of reducing a high-dimensional matrix to lower rank tensor networks, thereby interchanging costly matrix multiplication over full rank matrices, with sums over inner indices.

## 2.4 Skeleton Approximation

For this thesis, TCI was used, which we will discuss in the following section starting from a single two-dimensional matrix and calculating its skeleton decomposition [7]. It starts with a  $M \times N$  matrix  $A$  with all row indices being denoted as  $\mathbb{I}$  and all column indices as  $\mathbb{J}$ .  $A(\mathbb{I}, \mathbb{J})$  denotes the full matrix. To obtain an approximate rank  $D$  factorisation,  $D$  rows  $\{\mathcal{I}_1, \mathcal{I}_2, \dots, \mathcal{I}_D\} = \mathcal{I} \in \mathbb{I}$  and  $D$  columns  $\{\mathcal{J}_1, \mathcal{J}_2, \dots, \mathcal{J}_D\} = \mathcal{J} \in \mathbb{J}$  are selected, see fig. 2.4 [15]. Using a skeleton approximation this matrix can be decomposed into

$$A(\mathbb{I}, \mathbb{J}) \approx A_{\text{CI}}(\mathbb{I}, \mathbb{J}) = A(\mathbb{I}, \mathcal{J})[A(\mathcal{I}, \mathcal{J})]^{-1}A(\mathcal{I}, \mathbb{J}) \quad (2.26)$$

with the so-called pivot matrix  $A(\mathcal{I}, \mathcal{J})$ , which is formed by the elements contained in both  $\mathcal{I}$  and  $\mathcal{J}$  [16]. In contrast to SVD, where the full rank matrix has to be available, cross-interpolation (CI) allows us to generate an approximate with only a few elements of the exact matrix which removes memory limitations of having to store the full rank matrix. The CI has two properties:

1. The approximation is exact if  $A$  is of rank  $\leq D$  [15–18]. This is shown when calculating the determinant of an arbitrary matrix put into a  $2 \times 2$  block

$$\text{form } A' = \begin{pmatrix} A'_{11} & A'_{12} \\ A'_{21} & A'_{22} \end{pmatrix}$$

$$\det A' = \det[A'_{11}] \det[A'_{22} - A'_{21}A'_{11}^{-1}A'_{12}] \quad (2.27)$$

by introducing the Schur complement  $A'_{22} - A'_{21}A'_{11}^{-1}A'_{12}$ . Therefore, when adding an additional linearly dependent row and column to the pivot matrix  $A(\mathcal{I}, \mathcal{J})$  (which was chosen from linearly independent rows and columns) the determinant of the new pivot matrix

$$\det \begin{pmatrix} A(\mathcal{I}, \mathcal{J}) & A(\mathcal{I}, y_0) \\ A(x_0, \mathcal{J}) & A(x_0, y_0) \end{pmatrix} = 0 \quad (2.28)$$

vanishes. Thus, the determinant of the Schur complement, namely  $A(x_0, y_0) - A(x_0, \mathcal{J})[A(\mathcal{I}, \mathcal{J})]^{-1}A(\mathcal{I}, y_0)$ , must vanish too, showing that the reconstruction of the added row and column is exact.

2. The approximation is exact at the positions of the selected rows and columns  $i \in \mathcal{I}$  or  $j \in \mathcal{J}$  as the relation  $A(i, j) = A(i, \mathcal{J})A(\mathcal{I}, \mathcal{J})^{-1}A(\mathcal{I}, j)$  proves [15].

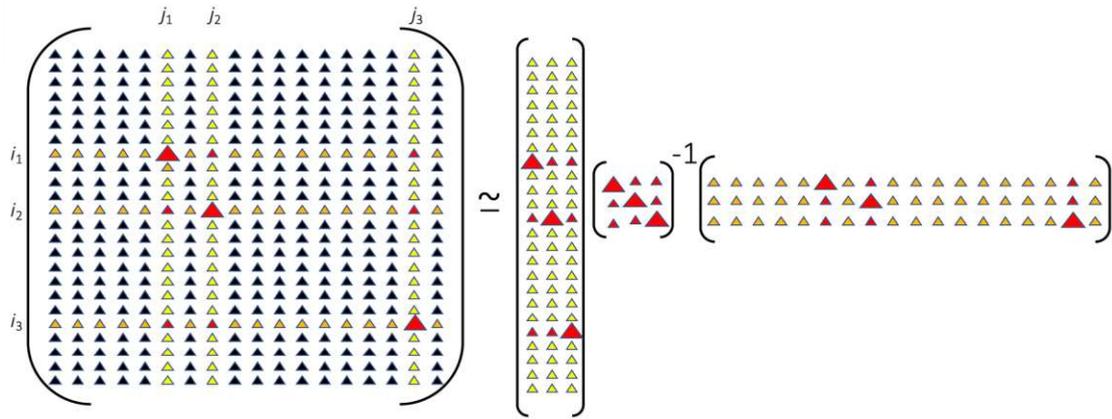


Figure 2.4: Visualisation of the skeleton approximation. The big red triangles are the pivots, the smaller red ones automatically generated pivots. Blue triangles are entirely interpolated by the sparse representation on the right [15].

The skeleton approximation is quasi-optimal meaning that, unlike SVD, it does not optimally compress a  $n \times n$  tensor. Therefore, the error is larger for the skeleton approximation [19] compared to a SVD of the same rank [4].

$$\|A - A_{\text{SVD}}^r\|_2 = \sqrt{\sigma_{r+1}^2 + \sigma_{r+2}^2 + \dots + \sigma_n^2} \quad (2.29)$$

$$\|A - A_{\text{TCI}}^r\|_2 \leq \sigma_{r+1} \sqrt{1 + r(n-r)} \quad (2.30)$$

$\|\cdot\|_2$  denotes the Euclidean norm with  $\sigma_{r+1}$  being the first singular value truncated for a SVD of rank  $r$ .

## 2.5 Tensor Cross Interpolation (TCI)

In the next step, the cross interpolation for a single matrix is generalised to  $n$ -dimensional tensors producing an iterative method that allows us to unfold a high-dimensional tensor into a tensor train (TT) [15, 16]. Although practically

not feasible, the following paragraph will give pedagogical insights into TCI before elaborating on a more practical algorithm.

A function with  $n$  discrete variables  $A(u_1, u_2, \dots, u_n)$ , each variable taking  $d$  different values, is depicted as a box with  $n$  legs sticking out of it, being the physical indices  $u_i$ , see fig. 2.5 left [15]. When regrouping the indices into one single index and a multi-index  $A_{(u_1),(u_2,u_3,\dots,u_n)}$  CI can be applied generating the pivot matrix (blue in fig. 2.5) and two sums over  $D_1$  hidden indices, which correspond to the  $\epsilon$ -rank of the CI (indicated by the green line in figure 2.5). As the tensor is reduced to a small amount of its original elements only a finite number of terms are summed up.

In a second step, the indices are once again regrouped into  $D_1$  indices of the pivot matrix and the second index  $u_2$  leading to  $A_{(u_1,u_2),(u_3,u_4,\dots,u_n)}$  which is then cross-interpolated adding another pivot matrix and two more green lines representing the sums over the inner indices. This procedure, one by one, unfolds the multi-index tensor into a tensor train, where each index is separated from the rest.

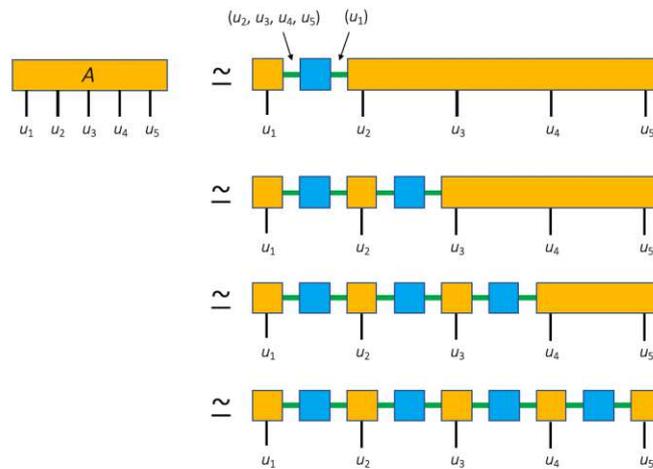


Figure 2.5: Abstract visualisation of constructing a TT. One by one the indices are grouped together at  $u_i$  and cross interpolated until the full matrix unfolds into a TT [15].

When translating the above into a practical algorithm with as few calls to the function  $A(u_1, u_2, \dots, u_n)$  as possible, keeping computation time and cost low, the

tensor is first split  $n - 1$  times to obtain the first pivot. When splitting the tensor at an arbitrary position  $\alpha$  with  $1 \leq \alpha \leq n$  it creates two sets of multi-indices  $i^{(\alpha)} = (u_1, u_2, \dots, u_\alpha)$  and  $j^{(\alpha)} = (u_\alpha, u_{\alpha+1}, \dots, u_n)$ . For each multi-index,  $D_\alpha$  subsets of multi-indices are considered for the CI, denoted as  $\mathcal{I}_\alpha = \{i_1, i_2, \dots, i_{D_\alpha}\}$  and  $\mathcal{J}_\alpha = \{j_1, j_2, \dots, j_{D_\alpha}\}$  of size  $\alpha$  and  $n - \alpha + 1$  respectively.

Example 1 elaborates on the method for a tensor of dimension  $d^4$  meaning that each variable  $u_\alpha$  can take  $d$  values. The tensor is split at position  $\alpha = 3$  creating multi-indices  $i^{(3)}$  and  $j^{(3)}$  containing  $\alpha = 3$  and  $n - \alpha + 1 = 4 - 3 + 1 = 2$  elements respectively. As a next step, the subsets  $\mathcal{I}_3$  and  $\mathcal{J}_3$  are chosen each containing  $D_3 = 2$  elements. Increasing  $D_\alpha$  means including more samples in  $\mathcal{I}_3$  and  $\mathcal{J}_3$ , thereby increasing the overall accuracy of the representation.

### Example 1

$$\begin{aligned}
 n = 4 \quad \alpha = 3 \quad D_3 = 2 \\
 i^{(3)} &= (u_1, u_2, u_3) \quad ; \quad j^{(3)} = (u_3, u_4) \\
 \mathcal{I}_3 &= \{(u_1, u_2, u_3)_1, (u_1, u_2, u_3)_2\} \\
 \mathcal{J}_3 &= \{(u_3, u_4)_1, (u_3, u_4)_2\}
 \end{aligned}$$

In order to construct the original tensor concatenation over the multi-indices is executed symbolised by  $\oplus$ , which means combining multiple matrices to form one large matrix, hence

$$A(u_1, \dots, u_{\alpha-1} \oplus u_\alpha \oplus u_{\alpha+1}, \dots, u_n) = A(u_1, u_2, \dots, u_n) \quad (2.31)$$

which is also valid for  $\mathcal{I}$  and  $\mathcal{J}$  with

$$\mathcal{I} \oplus \mathcal{J} = \{i \oplus j | i \in \mathcal{I}, j \in \mathcal{J}\}. \quad (2.32)$$

In example 2 the concatenation of two matrices is explicitly calculated.

## Example 2

$$\mathcal{I}_1 = \begin{Bmatrix} 0.1 & 0.2 \\ 0.3 & 0.4 \end{Bmatrix} \quad \mathcal{J}_2 = \begin{Bmatrix} 0.5 & 0.6 \end{Bmatrix}$$

$$\mathcal{I}_1 \oplus \mathcal{J}_2 = \begin{Bmatrix} \mathcal{I}_1 & \mathcal{I}_1 \\ \mathcal{J}_2 & \mathcal{J}_2 \end{Bmatrix} = \begin{Bmatrix} 0.1 & 0.2 & 0.1 & 0.2 \\ 0.3 & 0.4 & 0.3 & 0.4 \\ 0.5 & 0.5 & 0.6 & 0.6 \end{Bmatrix}$$

With these notations at hand two more tensors  $T_\alpha$  and  $P_\alpha$  are introduced corresponding to the (yellow) three-legged tensor and the (blue) Pivot matrix in fig. 2.5 respectively.

$$T_\alpha(i, u_\alpha, j) \equiv A(i \oplus u_\alpha \oplus j) \quad | \quad i \in \mathcal{I}_{\alpha-1}; j \in \mathcal{J}_{\alpha+1} \quad (2.33)$$

$$P_\alpha(i, j) \equiv A(i \oplus j) \quad | \quad i \in \mathcal{I}_\alpha; j \in \mathcal{J}_{\alpha+1} \quad (2.34)$$

Hence, the tensor  $T_\alpha$  is of dimension  $D_\alpha \times d \times D_{\alpha+1}$  and contracts over the physical index  $u_\alpha$  and two virtual indices, whereas  $P_\alpha$  is of dimension  $D_\alpha \times D_\alpha$ . Solely, the first and the last tensor  $T_1(i, u_1, j)$  and  $T_n(i, u_n, j)$  have dimension  $1 \times d \times D_\alpha$  and  $D_\alpha \times d \times 1$  respectively, hence being two-legged instead of three-legged. Furthermore, it is noteworthy that the elements of  $T_\alpha$  are constructed from slices of the original tensor  $A(\mathcal{I}_{\alpha-1}, u_\alpha, \mathcal{J}_{\alpha+1})$  along the physical index  $u_\alpha$ . Fig. 2.6 visualises the procedure for seven physical (black lines) and 12 hidden indices (green lines). For  $P_\alpha$  (blue boxes) the indices are transposed due to the inverse  $P_\alpha^{-1}$ . With the full unfolded TCI uncovered  $A$  can simply be expressed via matrix multiplication by

$$A(u_1, \dots, u_n) \approx A_{TCI}(u_1, \dots, u_n) \equiv \prod_{\alpha=1}^n T_\alpha(i, u_\alpha, j) P_\alpha^{-1}(i, j) \quad (2.35)$$

or more compact with  $i^{(\alpha)} \in \mathcal{I}_\alpha$  and  $j^{(\alpha)} \in \mathcal{J}_\alpha$

$$A_{TCI}(u_1, \dots, u_n) \equiv \prod_{\alpha=1}^n T_\alpha^{u_\alpha} P_\alpha^{-1} \quad (2.36)$$

$$= [T_1]_{1j^{(2)}}^{u_1} [P_1]_{j^{(2)}i^{(1)}}^{-1} [T_2]_{i^{(1)}j^{(3)}}^{u_2} \cdots [P_{n-1}]_{j^{(n)}i^{(n-1)}}^{-1} [T_n]_{i^{(n-1)}1}^{u_n}. \quad (2.37)$$

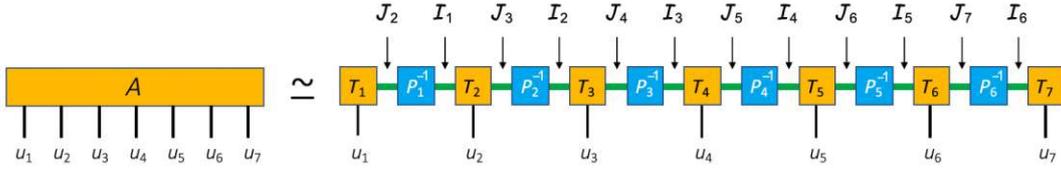


Figure 2.6: Tensor train for seven physical indices  $u_1, \dots, u_7$ , summed over a subset of rows ( $\mathcal{I}_\alpha$ ) and columns ( $\mathcal{J}_\alpha$ ), represented by matrix multiplication of  $T_\alpha$  and  $P_\alpha$  [15].

With an expression for the TCI established it is yet to be found how to select the rows and columns of  $\mathcal{I}$  and  $\mathcal{J}$  for a working algorithm.

### Nesting condition

The nesting condition [15, 16] restricts the selection of the pivots by constructing  $\mathcal{I}_\alpha$  ( $\mathcal{J}_\alpha$ ) from elements of  $\mathcal{I}_{\alpha-1}$  ( $\mathcal{J}_{\alpha+1}$ ). Thus, when defining  $\mathbb{I}_\alpha$  ( $\mathbb{J}_\alpha$ ) as the set of multi-indices  $u_\alpha$  of size 1 the relation arises

$$\mathcal{I}_\alpha \subset \mathcal{I}_{\alpha-1} \oplus \mathbb{I}_\alpha \quad (2.38)$$

$$\mathcal{J}_\alpha \subset \mathbb{J}_\alpha \oplus \mathcal{J}_{\alpha+1} \quad (2.39)$$

This ensures that for any multi-index  $i^{(\alpha-1)} \in \mathcal{I}_{\alpha-1}$  and  $j^{(\alpha+1)} \in \mathcal{J}_{\alpha+1}$ ,  $A_{\text{TCI}}(i^{(\alpha-1)} \oplus u_\alpha \oplus j^{(\alpha+1)})$  is exact [15]. Additionally, the approximation is exact, if the interpolation rank  $D_\alpha$  equals the rank of the original tensor  $A(i \oplus u_\alpha \oplus j)$  for any  $T_\alpha$  and  $(i, j)$  [16]. Example 3 demonstrates the nesting condition for a matrix with an arbitrary number  $n$  of physical indices  $\{u_1, u_2, \dots, u_n\}$  each taking 2 values.

### Example 3

$$\mathcal{I}_1 = \{(0.1), (0.2)\}$$

$$\mathcal{I}_2 = \{(0.1, 0.3), (0.2, 0.4)\}$$

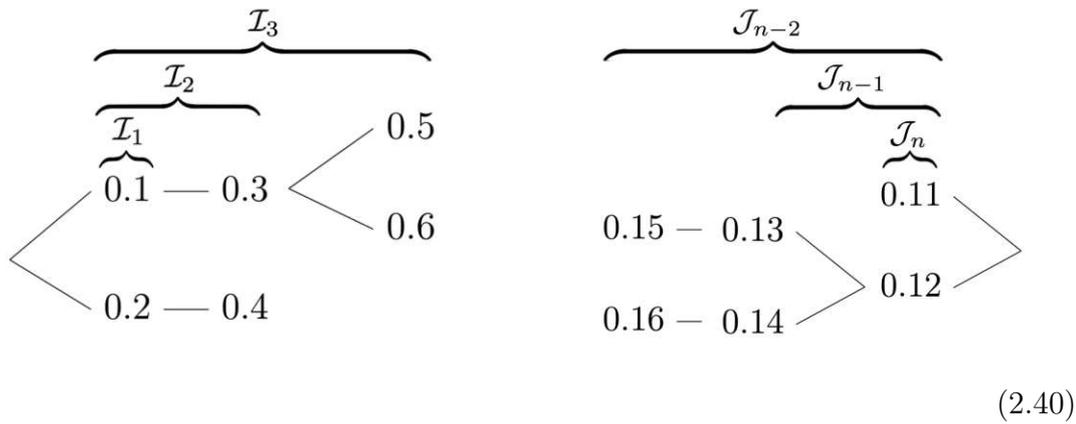
$$\mathcal{I}_3 = \{(0.1, 0.3, 0.5), (0.2, 0.3, 0.6)\}$$

$$\mathcal{J}_{n-2} = \{(0.12, 0.13, 0.15), (0.12, 0.14, 0.16)\}$$

$$\mathcal{J}_{n-1} = \{(0.12, 0.13), (0.12, 0.14)\}$$

$$\mathcal{J}_n = \{(0.11), (0.12)\}$$

For visualisation purposes, the pivots can be drawn as a tree diagram, see equ. 2.40. Expanding the tree gives further pivots, while keeping the nesting condition intact.



#### 2.5.1 Finding pivots

The accuracy and efficiency of the TCI depends heavily on the choice of pivots. It is critical to balance finding the best set of pivots as well as running an efficient algorithm [16]. In general, searching for the maximum-volume cross

$$\text{vol } A(\mathcal{I}, \mathcal{J}) = |\det A(\mathcal{I}, \mathcal{J})| \tag{2.41}$$

is expedient and efficient as the values included in the cross are exactly represented in the skeleton representation [16, 20]. Another method is introducing the error

tensor  $\Pi$  to build a MPS via TCI [15]

$$\Pi_{\alpha}^{u_{\alpha}u_{\alpha+1}} \equiv A(\mathcal{I}_{\alpha-1} \oplus \mathbb{1}_{\alpha} \oplus \mathcal{J}_{\alpha+1} \oplus \mathcal{J}_{\alpha+2}) \quad (2.42)$$

$$\epsilon_{\Pi}(i^{(\alpha-1)}, u_{\alpha}, u_{\alpha+1}, j^{(\alpha+1)}) \equiv \left| \Pi_{\alpha}^{u_{\alpha}u_{\alpha+1}} - [T_{\alpha}]_{i^{(\alpha-1)} k}^{u_{\alpha}} [P_{\alpha}]_{k l}^{-1} [T_{\alpha+1}]_l^{u_{\alpha+1} j^{(\alpha+2)}} \right|. \quad (2.43)$$

with  $\Pi_{\alpha}^{u_{\alpha}u_{\alpha+1}}$  being a four-legged tensor with two physical indices  $u_{\alpha}$  and  $u_{\alpha+1}$ , see fig. 2.7. Due to the nesting condition introduced above the error function in equation 2.42 is equivalent to

$$\epsilon_{\Pi}(i^{(\alpha-1)}, u_{\alpha}, u_{\alpha+1}, j^{(\alpha+2)}) = |A - A_{\text{TCI}}|(i^{(\alpha-1)}, u_{\alpha}, u_{\alpha+1}, j^{(\alpha+2)}) \quad (2.44)$$

meaning that improving the choice of pivots for a single factorisation improves the overall TCI [15]. With the error function at hand the algorithm can proceed

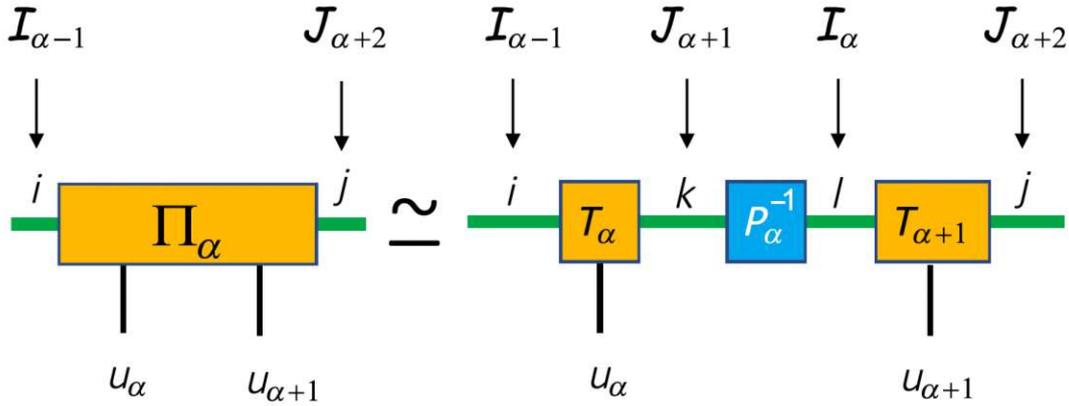


Figure 2.7: Error function on the left stems from the TCI. Green lines represent sums over indices, black lines physical indices [15].

finding its first pivot with  $D_{\alpha} = 1$  adding the new found crosses to  $\mathcal{I}_{\alpha}$  and  $\mathcal{J}_{\alpha+1}$  before moving to the next site  $\Pi_{\alpha+1}^{u_{\alpha+1}u_{\alpha+2}}$ . Pivots are chosen to maximise the error function, in that, by adding values that produce the maximum error to  $\mathcal{I}$  and  $\mathcal{J}$  these elements become exact, minimising the overall error by the greatest extent possible. This method is known under the name of "Active Learning" in machine learning [21]. The concept allows the algorithm to choose the data from which it learns by itself and therefore provides a better performance with less training. In this fashion, the algorithm adds pivots for each site completing one "sweep"

when reaching the very last site  $\Pi_{n-1}^{u_{n-1}u_n}$ . The process is reversed sweeping backwards from the very last site to the first one. Thus, by sweeping back and forth more pivots are successively added and the approximation is enhanced while also keeping the nesting condition intact [16]. The construction of the TCI is aborted when either the error falls below a desired tolerance, after which, as a precaution, two more sweeps are performed, to avoid any local minima, or the number of sweeps exceeds a predefined maximum.

For finding a suitable pivot either the full-sized tensor  $\Pi_{\alpha}^{u_{\alpha}u_{\alpha+1}}$  is searched which would result in a brute force search over  $(D_{\alpha}d)^2$  elements (*full search*). For a more time-efficient method the error function is iteratively improved scanning  $(i^{(\alpha-1)}, u_{\alpha})$  and  $(u_{\alpha+1}, j^{(\alpha+2)})$  alternatively until a local maximum is found or the number of iterations exceeds a bound value. This would correspond to  $\mathcal{O}(D_{\alpha}d)$  elements searched (*rook pivoting*) [15].

## 2.5.2 Quantics Tensor Cross Interpolation (QTCI)

Quantics tensor cross interpolation combines two methods highly efficient in describing many-body problems, quantics tensor trains (QTT) and TCI [5]. As the latter was already described above, the subsequent section will elaborate on QTT. The main idea of QTT is that instead of using a standard equidistant grid to discretise function variables, exponentially different length scales are introduced [22]. With a single variable  $u_{\alpha}$  decomposed into a set of binary variables only taking  $\sigma_{\alpha,i} \in \{0, 1\}$ , the function  $f(u_{\alpha}) = f(\sigma_{\alpha,1}, \sigma_{\alpha,2}, \dots, \sigma_{\alpha,R}) = f_{\sigma}$  reduces to a  $2 \times \dots \times 2$  ( $R$ -way) tensor, with  $R$  depending on the grid-size  $2^R$ . Example 4 shows the quantisation of two different  $u_{\alpha}$ .

### Example 4

$$R = 5$$

$$f(0) \rightarrow f(0, 0, 0, 0, 0)$$

$$f(3) \rightarrow f(0, 0, 0, 1, 1)$$

In example 5, the binary variables or *qubits* of same length scale are grouped

together when building the tensor train [6]. This allows separating different length scales when weakly entangled by compressing the corresponding tensor or embracing the entanglement where needed by increasing the tensor's rank. This becomes also essential when adding dimensions  $n$  as the entanglement is expected to be largest for two qubits of the same length scale.

### Example 5

$$\begin{aligned}
 R = 3 \quad n = 2 \\
 f(*, 1) &\rightarrow f(*, 0, *, 0, *, 1) \\
 f(5, *) &\rightarrow f(1, *, 0, *, 1, *) \\
 f(5, 1) &\rightarrow f(1, 0, 0, 0, 1, 1)
 \end{aligned}$$

Unfolding the quantised tensor into a QTT gives the equation:

$$f_{\sigma} = \prod_{\alpha=1}^{nR} M_{\alpha}^{\sigma_{\alpha}} = [M_1]_{1\alpha_1}^{\sigma_1} [M_2]_{\alpha_1\alpha_2}^{\sigma_2} \dots [M_{nR}]_{\alpha_{(nR-1)}1}^{\sigma_{nR}}. \quad (2.45)$$

Note that identical indices imply the sum over that index (Einstein notation). Similarly as before,  $\sigma_{\alpha}$  corresponds to the physical indices and  $\alpha_i$  to the virtual (inner) indices. Equation 2.45 is the exact representation of function  $f_{\sigma}$  reshaped into a tensor train. Its bond dimension increases exponentially with each site  $\alpha$  towards the chain's centre  $D'_{\alpha} = 2^{\min\{\alpha, nR-\alpha\}}$  [5]. This exact decomposition (*full rank*) is visualised in fig.2.8.

For certain systems it is beneficial to group indices together [5]. *Fusing* indices that are expected to have high levels of entanglement saves the cost of compressing a barely compressible tensor, but instead resources can be focused on compressing less entangled indices. When dealing with  $n$  dimensional systems, it suggests itself fusing indices of the same length scale. For QTT this means that instead of 2 values each index can take  $2^n$  values. Thereby, the number of outer indices decreases by a factor of  $1/n$  meaning that for the QTT only  $R$  arguments have to be taken into

account, visually corresponding to a TT with  $R$  legs sticking out.

$$\boldsymbol{\sigma}_\alpha = (\sigma_{\alpha,1}, \sigma_{\alpha,2}, \dots, \sigma_{\alpha,R}) \mid \sigma_{\alpha,i} \in \{0, 1\} \quad (2.46)$$

$$\tilde{\sigma}_\alpha = \sum_{i=1}^n 2^{i-1} \sigma_{\alpha,i} \in \{0, \dots, 2^n - 1\} \quad (2.47)$$

Example 6 calculates the fused representation for a simple 3D system. Instead of taking only two values  $\sigma_\alpha \in \{0, 1\}$ , one variable may take  $\tilde{\sigma}_\alpha \in \{0, \dots, 2^n - 1\}$  when fusing  $n$  indices  $\{u_1, u_2, \dots, u_n\}$  together. For the right slot  $f(*, *, u_3)$ , in the first line, the binary representation remains the same taking values  $\{0, 1\}$ . Moving on the middle index  $f(*, u_2, *)$ , in the second line, the binary variables takes the two values  $\{0, 2\}$ , and the index on the left  $f(u_1, *, *)$ , in the third line takes values  $\{0, 4\}$ . Simply taking the sum of all dimensions gives the fused representation.

### Example 6

$$\begin{array}{lll} R = 3 & n = 3 & \tilde{\sigma}_\alpha \in [0, 7] \\ f(0, 0, 3) & \rightarrow & f(0, 1, 1) \\ f(0, 1, 0) & \rightarrow & f(0, 0, 2) \\ f(5, 0, 0) & \rightarrow & f(4, 0, 4) \\ f(5, 1, 3) & \rightarrow & f(4, 1, 7) \end{array}$$

For the algorithm, the function variables  $u_\alpha$  have to be normalised to lie in  $[0, 1]$  before being discretised to the  $2^R$  grid [5]. In example 7 the function variable  $u_1$  is normalised before being transformed into its qubits representation.

### Example 7

$$\begin{array}{ll} R = 5 & u_1 \in [0, 2\pi) \\ f(0) & \rightarrow f(0, 0, 0, 0, 0) \\ f(2\pi/2^5) & \rightarrow f(1/2^5) \rightarrow f(0, 0, 0, 0, 1) \end{array}$$

To achieve compression and save computational cost TCI has to enter the equation. In this way dominant features can be still accurately reproduced, whereas less important contributions are truncated. When performing TCI

equation 2.37 changes to

$$f_{\sigma} = \prod_{\alpha=1}^{nR} T_{i_{\alpha}, j_{\alpha}}^{\sigma_{\alpha}} P_{i_{\alpha}, j_{\alpha}}^{-1} \quad (2.48)$$

$$= [T_1]_{1j^{(2)}}^{\sigma_1} [P_1]_{j^{(2)}i^{(1)}}^{-1} [T_2]_{i^{(1)}j^{(3)}}^{\sigma_2} [P_2]_{j^{(3)}i^{(2)}}^{-1} \cdots [T_{nR}]_{i^{(nR-1)}1}^{\sigma_{nR}} \quad (2.49)$$

with  $i^{(\alpha)} \in \mathcal{I}_{\alpha}$  and  $j^{(\alpha)} \in \mathcal{J}_{\alpha}$  representing the pivot crosses. Although considerably truncated towards the centre, bond dimension  $D_{\alpha}$  is expected to grow exponentially in the beginning as on large length scales there are no details to spare without losing a considerable amount of information, following  $D_{\alpha} = D'_{\alpha}$  [6]. When reaching a maximum bond dimension  $D_{\max}$ , the entanglement between different length scales peaks leading to more and more features being compressed,  $D_{\alpha} \ll D'_{\alpha}$ . In general, the bond dimension reaches a plateau after  $D_{\max}$  or slowly starts to decrease, as entanglement decreases. If finer length scales do not yield any additional information, say entanglement between different length scales vanishes, the bond dimension drops to 2, which can be represented as a product state, dropping the sums over the inner indices [4]. Maximum bond dimension is closely related to the overall error, thus increasing bond dimension allows for describing more complex physical systems with intricate structure [14].

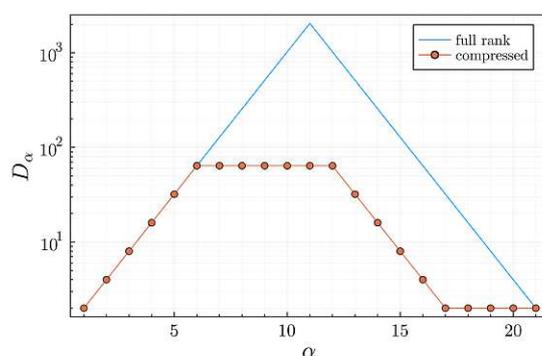


Figure 2.8: Exemplary bond dimensions for a QTCI. For  $\alpha \leq 6$  exponential growth (not compressible),  $\alpha \in [7, 12]$  constant  $D_{\alpha}$  (compressible),  $\alpha \in [13, 17]$  exponential decrease (compressible, entanglement diminishes),  $\alpha > 17$   $D_{\alpha} = 2$  (no entanglement).

Besides TCI there are other ways to achieve a compressed tensor train, with SVD being one of the most commonly used methods. By truncating all singular

values smaller than a predefined truncation threshold, SVD optimally suppresses its maximum bond dimension  $D_{\max}$  for a given tolerance  $\epsilon$  whereas TCI is quasi-optimal by finding a local minimum for  $D_{\max}$  (generally being slightly higher than for SVD) [5, 15]. SVD makes use of the full rank tensor to find an optimal solution which consequentially involves exponentially long run time even if the tensor is highly compressible. In contrast, TCI does not require evaluating or storing the full rank tensor which gives it a major advantage over SVD when it comes to memory usage and computation time as TCI constructs the TT from a few sampled values. As for the numerical effort, SVD requires  $\mathcal{O}(D_{\max}^2 n R D'_\alpha)$  values to be calculated [4] with  $D'_\alpha$  being the uncompressed tensor's dimension of site  $\alpha$  whereas TCI needs  $\mathcal{O}(D_{\max}^2 n R)$  [5].

## 3 Results and Discussion

All calculations of the QTCI were performed in julia 1.9 using the package presented in the paper from Ritter et al. (2023) [5]. It is important to note that julia uses 1-based indexing. In order to align with the programming language the library uses bits  $\{1, 2\}$ , instead of  $\{0, 1\}$ . Thus, the bit representation of example 5 changes to

### Example 8

$$\begin{aligned}
 R = 3 \quad n = 2 \\
 f(*, 2) &\rightarrow f(*, 1, *, 1, *, 2) \\
 f(6, *) &\rightarrow f(2, *, 1, *, 2, *) \\
 f(6, 2) &\rightarrow f(2, 1, 1, 1, 2, 2)
 \end{aligned}$$

For simple calculations an Apple M2 processor was used. High-demanding calculations were performed on the Vienna Science Cluster (VSC) [23].

### 3.1 One dimension

In the following chapter the dispersion relation of a 1D tight-binding model will be studied with the band dispersion

$$\epsilon(k) = -2t \cos(k) - \mu \quad (3.1)$$

and a self-energy  $\Sigma$  inspired by self-energy deep in the Mott phase [24] of

$$\Sigma(k, \omega) = U/2 + \frac{U^2}{4} \frac{1}{i\omega_n + \alpha\epsilon(k)} \quad (3.2)$$

The parameters  $U$ ,  $t$  and  $\alpha$  were set to 4.0, 2.0 and 1.0 respectively.

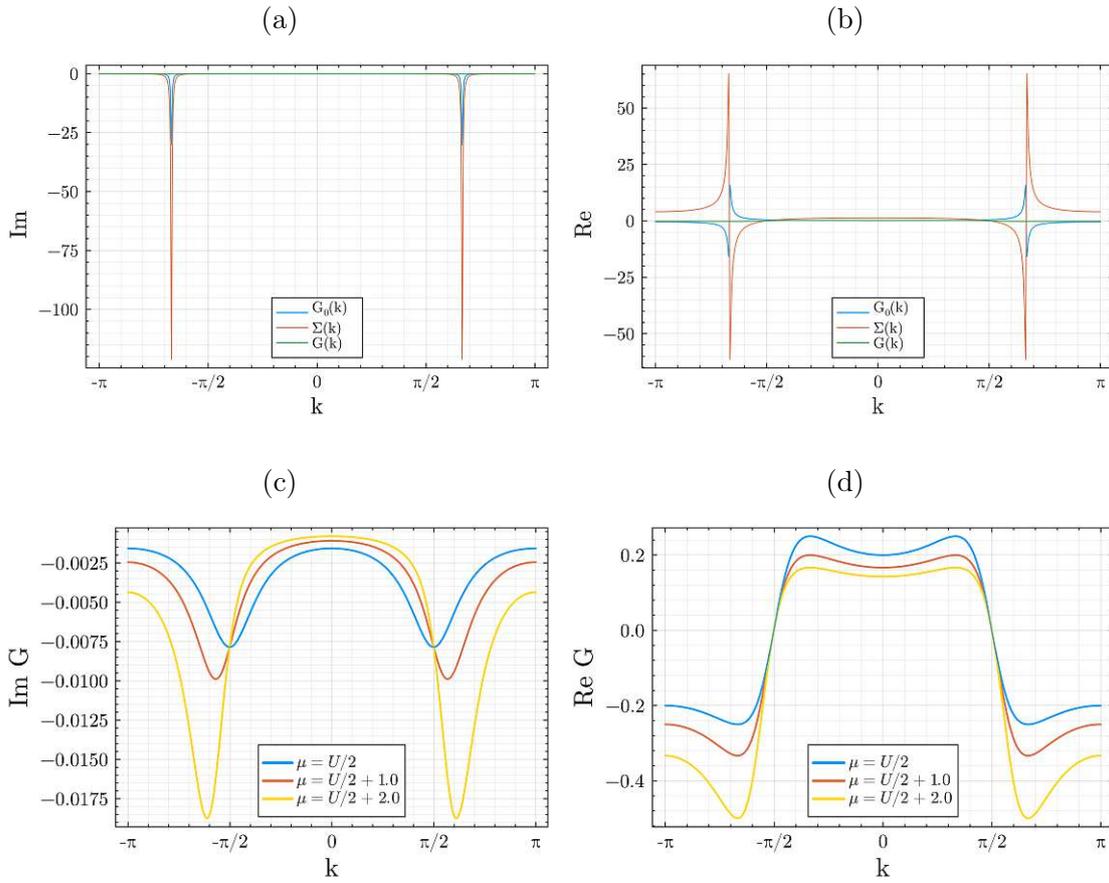


Figure 3.1: Dependence of the (a) imaginary and (b) real part of  $G_0$ ,  $\Sigma$ ,  $G$  on  $k$  in the first Brillouin zone for  $\omega_0 = \pi/\beta$  and  $\beta = 100$  at half-filling  $\mu = U/2$ . Panels (c) and (d) show  $G$  for different fillings.

Figures 3.1(a,b) depict the real and imaginary part of the non-interacting Green's function  $G_0(k, \omega)$ , self-energy  $\Sigma(k, \omega)$  and the interacting Green's function  $G(k, \omega)$  for the lowest positive Matsubara frequency  $\omega_0 = \pi/\beta$  and  $\beta = 100$  at half-filling  $\mu = U/2$ . In fig. 3.1(a,b) the curves of  $\Sigma$  and  $G_0$  show distinct peaks for the Fermi point, thus requiring a large grid size to accurately display them (here  $2^{10}$  sampling points). For half-filling the extrema of  $G$  are located at  $k = \pm\pi/2$ . The structure of  $G$  is considerably flatter for half-filling than for  $G_0(k, \omega)$ ,  $\Sigma(k, \omega)$ , meaning that it is also easy to compress. However, when moving away from half-filling  $\mu > U/2$  the function changes rapidly, and sharp peaks emerge, see fig. 3.1(c,d). When further increasing  $\mu$  the structure of

the function becomes more complex and therefore, less compressible when building the QTCI, cf. fig. 3.11b below.

Fig. 3.2 shows the dependence of the self-energy  $\Sigma(i\omega_n)$  for different  $k$ -points, along high symmetry lines of a cubic lattice, and for different fillings. Using the imaginary part of  $\Sigma(i\omega_n)$ , the scattering rate (inverse life time) of the particle can be determined, whereby the material can be characterised as an insulator (fig. 3.2(a) for  $\vec{k}(\pi, 0)$  and  $\vec{k}(\pi/2, \pi/2)$  with  $\text{Im } \Sigma(i\omega_n) \neq 0$  for all  $n \in \mathbb{R}$ ) or a metal ( $\text{Im } \Sigma(i\omega_n) = 0$ ).

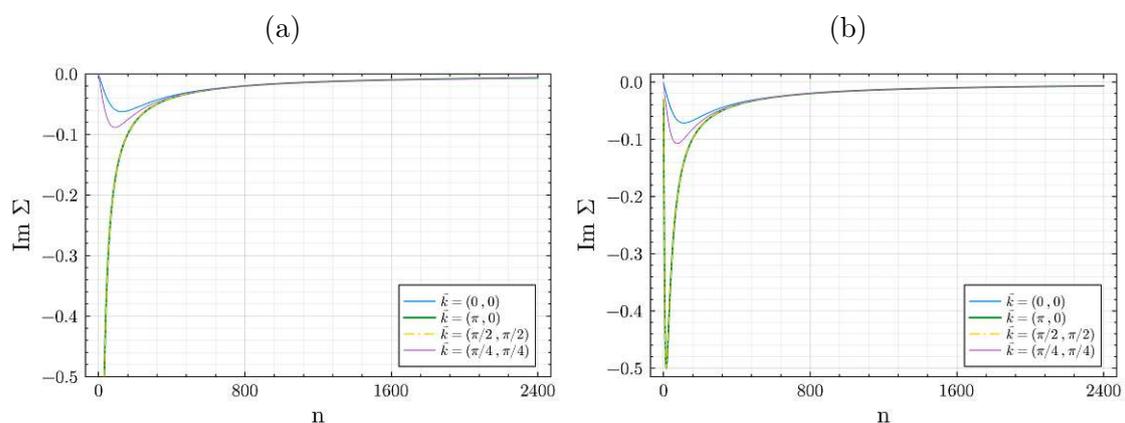


Figure 3.2: Dependence of  $\text{Im } \Sigma$  on  $i\omega_n$  at different  $k$ -points for  $\omega_0 = \pi/\beta$  and  $\beta = 100$ , (a)  $\mu = 0$ , (b)  $\mu = U/2$  (half-filling)

Fig. 3.3 compares the Green's function  $G$  for different  $\beta$  and frequencies at half-filling  $\mu = U/2$ . At half-filling the structure of the function is hardly affected when decreasing the temperature or increasing the frequency but the pole at  $k = \pm\pi/2$  becomes sharper. However, when moving away from half-filling  $\mu > U/2$  we dope the Mott insulator and the system becomes metallic. Here, the effect of temperature steadily increases, making the structure of  $G$  more intricate with high and sharp peaks, see fig. 3.4. Therefore,  $G$  outside half-filling requires a larger grid size to be accurately resolved, which is also reflected in the bond dimensions when calculating the QTCI, see fig. 3.5.

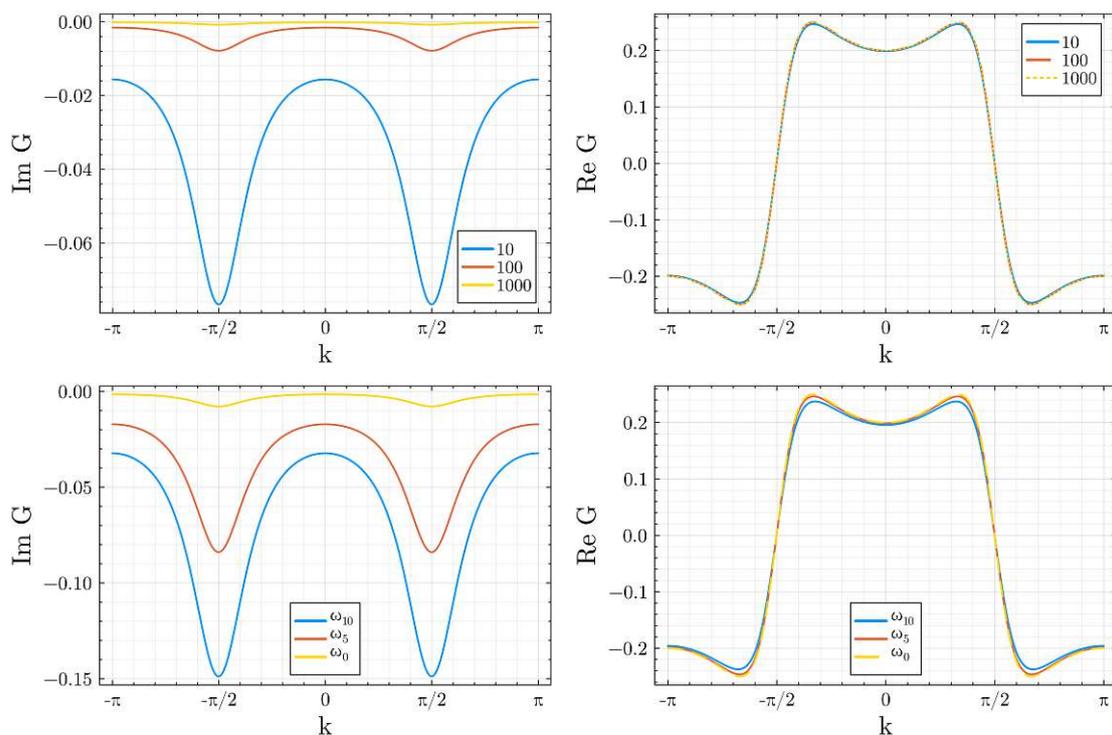


Figure 3.3: Imaginary part (left) and real part (right) of  $G(k, \omega_n)$  at half-filling  $\mu = U/2$  for different inverse temperature  $\beta = \{10, 100, 1000\}$ , for  $\omega_0$  and for different frequencies  $\omega_n = \{\omega_0, \omega_5, \omega_{10}\}$  and  $\beta = 100$  (bottom). Increasing temperature has little effect on the function at half-filling.

In the next step the QTCI was calculated. The bond dimension of the TT is a useful quantity to study the compressibility of the tensor. As a reference, the bond dimension of the full rank tensor (blue) is plotted as well referring to an

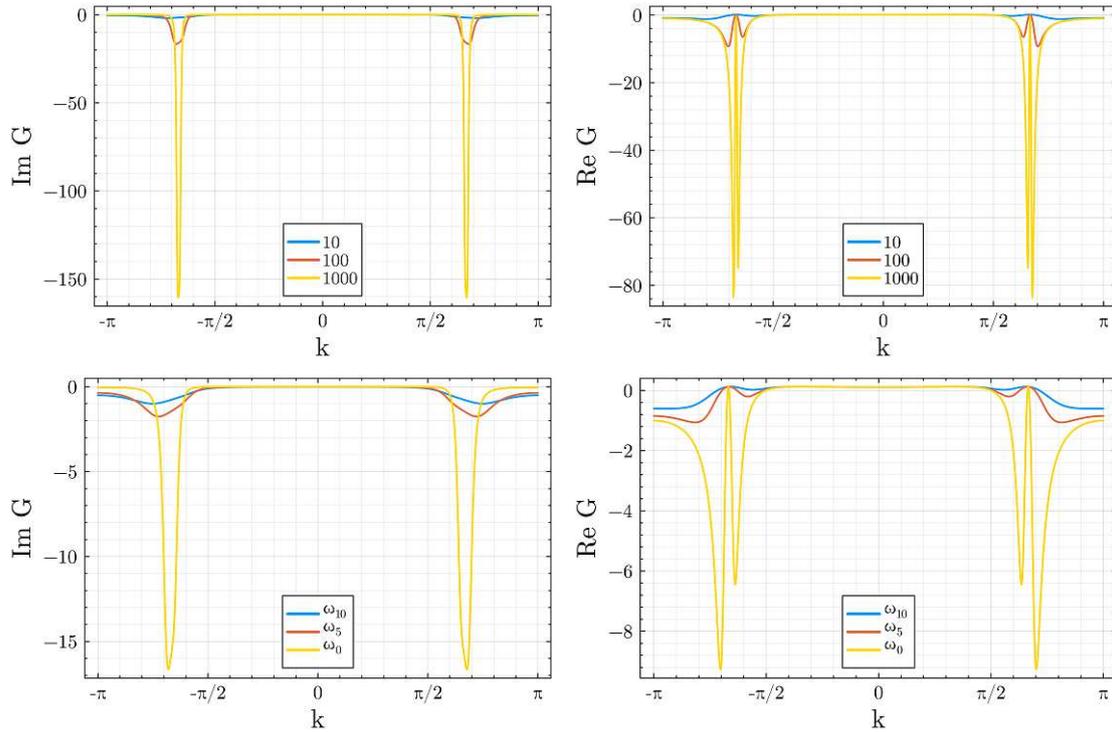


Figure 3.4: Imaginary part (left) and real part (right) of  $G(k, \omega_n)$  outside of half-filling  $\mu = U/2+4$  for different inverse temperature  $\beta = \{10, 100, 1000\}$ ,  $\omega_0$  and different frequencies  $\omega_n = \{\omega_0, \omega_5, \omega_{10}\}$ ,  $\beta = 100$  (bottom). Decreasing temperature creates intricate structures with high and sharp peaks that need a larger grid to be accurately represented.

incompressible data set. Figure 3.5 shows the bond dimension for the QTCI of  $G$  with  $R = 24$  equalling  $2^{24}$  points on an equidistant grid and a maximum error tolerance of  $10^{-8}$ . Although  $2^{24}$  sampling points are unnecessary, choosing  $R$  to be larger than required provides an excellent opportunity to test the methods ability to compress the data adequately. The plot compares different inverse temperatures, fig. 3.5(a), and orders of magnitude of the Matsubara frequency, fig. 3.5(b). As expected the bond dimension grows exponentially in the beginning meaning that different length scales in this range are highly entangled and not compressible. The maximum bond dimension depends both on  $\beta$  and  $\omega_n$  but either way stays well below the full rank of  $2^{12}$  bonds. After reaching a maximum, the bond dimensions start to decrease showing that finer length scales tend to be less entangled and thus compressible. As depicted in fig. 3.5(c) off

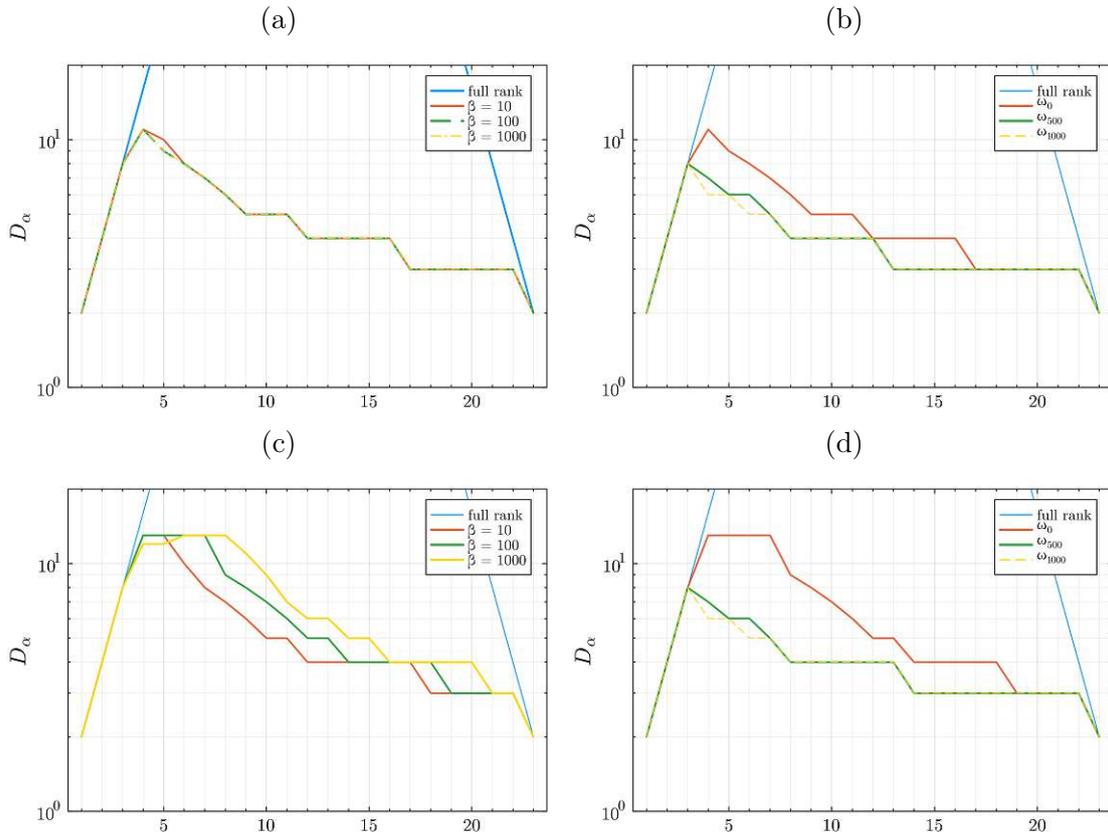


Figure 3.5: Bond dimension for QTCI of  $G(\mathbf{k}, \omega_n)$  at half-filling  $\mu = U/2$  (a,b) and out of half-filling  $\mu = U/2 + 4$  (bottom) for varying (a,c) inverse temperature with  $\omega_0$ , (b,d) order of magnitude of the Matsubara frequency with  $\beta = 100$  fixed. The blue line represents the full rank (exact) representation of the MPS.

half-filling ( $\mu > U/2$ ) lower temperatures require slightly higher bond dimensions for an accurate display within the given error. From this follows that calculations in the limit  $T \rightarrow 0$  will be computationally more demanding due to its finer structures. Notably, the maximum bond dimension  $D_{\max}$  stays constant when decreasing the temperature. Similarly, for larger frequencies the maximum bond dimension is reduced, see fig. 3.5(b,d). For high orders of magnitude of the Matsubara frequency, the information gain is minimal as demonstrated by the bond dimensions for  $\omega_{500}$  and  $\omega_{1000}$  being virtually identical.

### 3.1.1 Error of QTCI

The error of the QTCI is determined dynamically by comparing the result of one sweep with the tensor calculated in the sweep before. If the relative error falls below a fixed tolerance  $\epsilon$  two more sweeps are performed to rule out a local minimum before moving on to the next site. Fig. 3.6 shows the error for  $G$  for  $\beta = 100$ ,  $\mu = U/2$  and a tolerance of  $\epsilon = 10^{-8}$ . The distinct spikes in fig. 3.6(b) at  $k = \pm\pi/2$  are due to  $|G|$  falling near zero thus increasing the relative error abruptly. Overall, the error closely matches the predefined tolerance  $\epsilon$  when comparing the function reconstructed from QTCI with the original one. This proves that the method to improve the QTCI as well as to estimate the error is reliable. Notably, the error exhibits a periodicity, displaying 64 peaks in the interval  $[0, 2\pi)$ , which might stem from the separation of length scales.

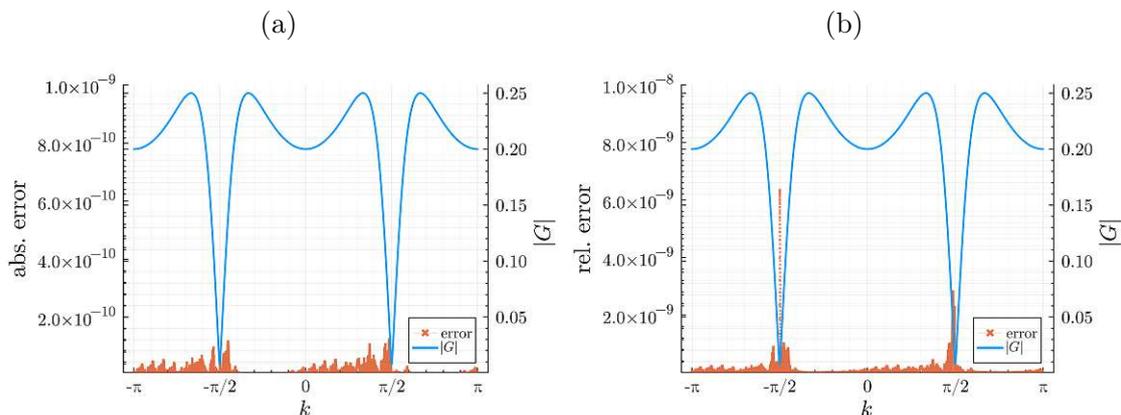


Figure 3.6: (a) Absolute and (b) relative error for the reconstructed values of the QTCI for  $G(i\omega_0, k)$  with  $\beta = 100$ ,  $\epsilon = 10^{-8}$ .

### 3.1.2 Comparison of SVD and QTCI

Comparing bond dimension from QTCI with SVD for  $G(\beta = 100, i\omega_0, \mu = 0)$  confirms the conjecture that SVD compresses slightly more efficiently, see fig. 3.7. It allows for an optimal compression by considering the whole tensor whereas QTCI comes to a quasi-optimal result by iteratively building the interpolation with single rows and columns of the tensor, see chapter 2.4. This means using SVD instead of QTCI comes at the cost of increased memory demand and computation time but

lowers the bond dimensions as demonstrated by fig. 3.7.

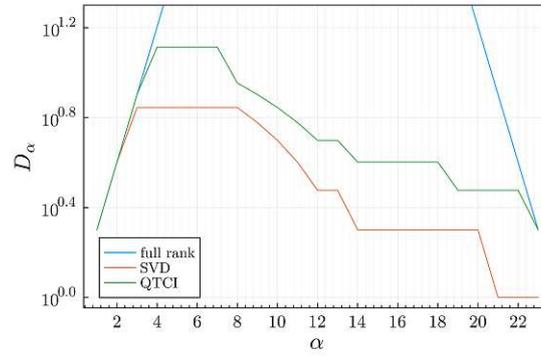


Figure 3.7: MPS built with SVD requires less bond dimensions, thus providing an optimal compression, whereas QTCI requires less memory and computation time.

## 3.2 Two dimensions

In two dimensions the band dispersion becomes

$$\epsilon(\mathbf{k}) = -2t_x \cos(k_x) - 2t_y \cos(k_y) - \mu. \quad (3.3)$$

The relation simplifies for  $t_x = t_y = t$  which refers to an isotropic model.

Fig. 3.8 depicts the real and imaginary part of  $G$ ,  $\Sigma$ ,  $G_0$  in two dimensions for  $\beta = 100$ ,  $\omega_0$  and  $\mu = U/2$ . Function values vary little except in the region of the Fermi surface where the function's extrema are located. The imaginary parts of the functions are negative throughout the first Brillouin zone thus preserving causality. When comparing the curve of  $G$  to  $\Sigma$  and  $G_0$  it becomes apparent, that the Fermi surface deforms when including electronic interaction. Furthermore, the peaks broaden similar to scenarios where temperature is increased, cf. fig. 3.9, as the interaction makes scattering of the propagating electron more likely.

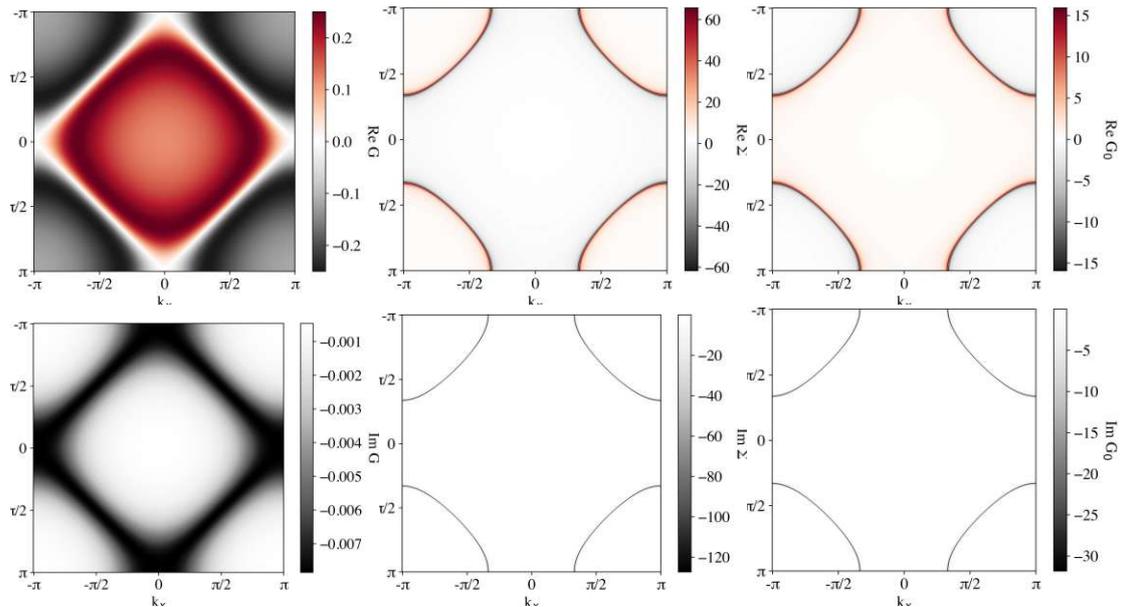


Figure 3.8: Heatmaps of the real (top) and imaginary part (bottom) for  $G$ ,  $\Sigma$  and  $G_0$  at  $\beta = 100$ ,  $\mu = U/2$  and  $\omega_0$ . Adding or removing electrons is only possible at low frequencies near the Fermi surface, cf. fig. 2.2.

When increasing the temperature, the structure becomes less intricate due to higher scattering rate (lower life time) of the propagating particle and the peaks

of the Fermi surface broaden as depicted in fig. 3.9.

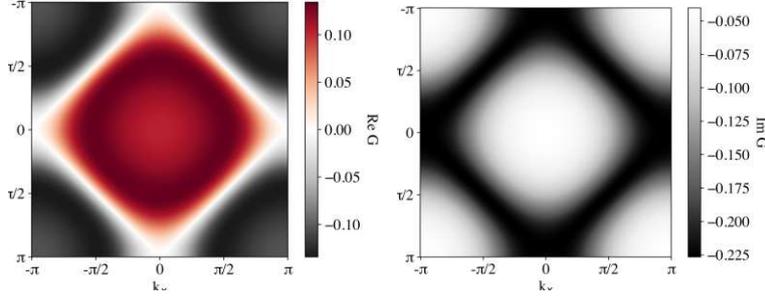


Figure 3.9: Heatmaps of the real (left) and imaginary part (right) for  $G$  at  $\beta = 1$ . The Fermi surface broadens when increasing temperature.

Fig. 3.10 shows the path through the high-symmetry points within the first Brillouin zone of a square lattice. Due to  $G$  being much smaller than  $G_0$  or  $\Sigma$  the function values are depicted on the right  $y$ -axis. For half-filling  $G$  is constant around the Fermi surface. The symmetry of the structure allows for nested wave-vectors meaning that  $\tilde{\epsilon}(\mathbf{k}) = \epsilon(\mathbf{k} + \mathbf{q})$  with a constant wave vector  $\vec{q}$ . This gives rise to instabilities of the system producing distinct material properties, like spin density waves. However, this kind of physics is not included in our self-energy equation 3.2.

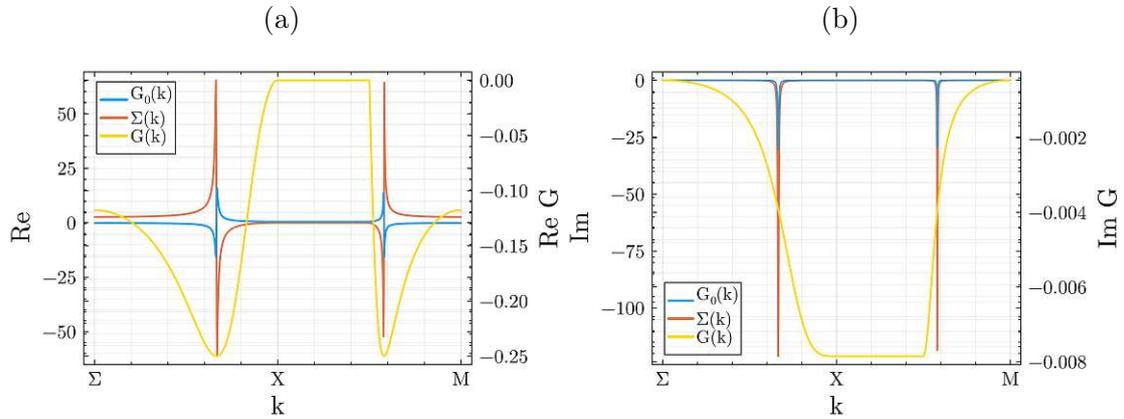


Figure 3.10: Path through the high symmetry points of a cubic lattice with the (a) real and (b) imaginary part.

Building QTCI for the two-dimensional model approximately increases the bond dimension tenfold ( $D_{\max} \approx \{15; 130\}$ ) for  $R = 24$ ,  $\epsilon = 10^{-8}$  and the lowest

positive Matsubara frequency  $\omega_0$  compared to one dimension, see fig. 3.11. As in the previous case the bond dimension stays significantly lower than the fully entangled (exact) limit TT depicted in blue in fig. 3.11, thus once more reducing the computational cost decisively. As before the bond dimension scales with the inverse temperature outside half-filling  $\mu = U/2 + 4$ , meaning that the computational cost increases with  $\beta$ , see fig. 3.11b, but stays stable for half-filling  $\mu = U/2$ , see fig. 3.11a.

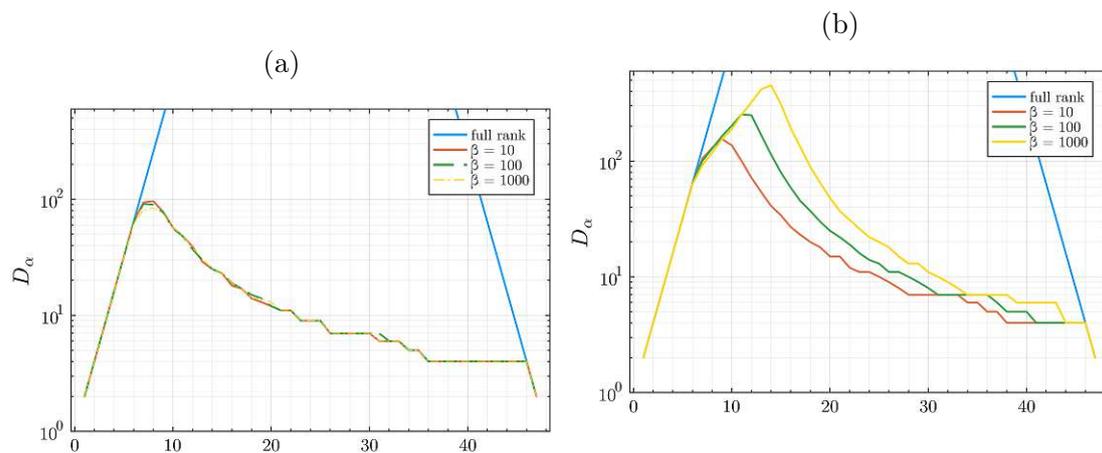


Figure 3.11: Bond dimension for the QTCI of  $G$  for different inverse temperature  $\beta$  at (a) half-filling  $\mu = U/2$ , (b) for  $\mu = U/2 + 4$ .

### 3.2.1 Error of QTCI

The error range of the QTCI in two dimensions is comparable to one dimension when keeping to  $\epsilon = 10^{-8}$ , as depicted in fig. 3.12 for  $G$  at  $\beta = 100$ ,  $\mu = U/2$ . Noticeable are the white spots around the Fermi surface, exhibiting a smaller error than the rest of the space. This is likely due to choosing pivots containing large values for building the QTCI, thereby, decreasing the overall error the most. In consequence, these values become exact, see chapter 2.5, paragraph **Nesting Condition**. This suggests that the algorithm is focusing on high resolution in regions with more intricate structures whereas in monotonic regions a higher error is tolerated.

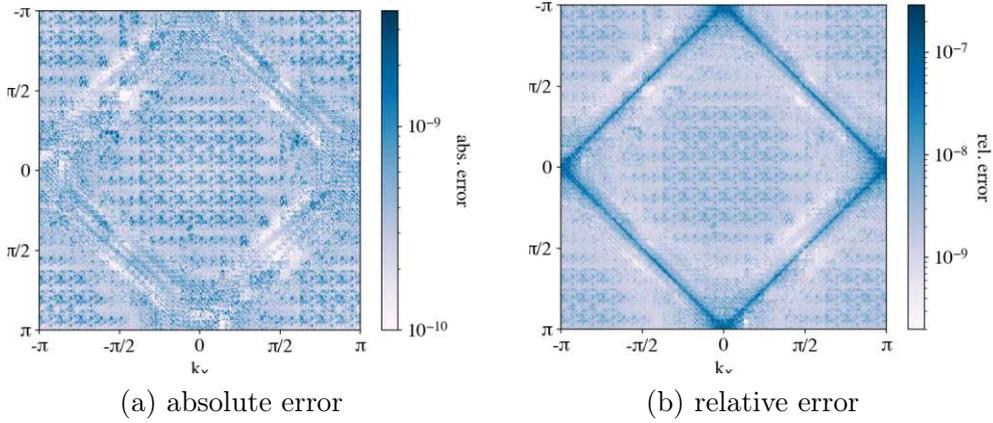


Figure 3.12: Error for the QTCI of  $G$  for  $R = 24$ ,  $\beta = 100$  and  $\mu = U/2$ . The minimum of the absolute error function is at the Fermi surface where values of  $G$  are largest.

### 3.2.2 Effect of tolerance $\epsilon$ on maximum bond dimension

$$D_{\max}$$

Building a MPS from QTCI naturally comes along with interpolating data, thereby accepting a certain level of error. For QTCI this means limiting the bond dimensions and suppressing numerically negligible entanglement between different length scales. When providing a more accurate MPS and decreasing the error, bond dimensions have to be increased. Fig. 3.13(a) demonstrates the dependence of the bond dimension for each site  $\alpha$  on the tolerance  $\epsilon$ . The algorithm calculates  $\epsilon$  by comparing QTCI from the previous sweep with the QTCI of the current sweep, and terminates the execution (after two additional sweeps) when the difference of the two QTCI drops below  $\epsilon$ . Although  $\epsilon$  does not directly correspond to the absolute error of the QTCI compared to the original function, it gives a practical tool to tune accuracy and complexity of the MPS. Fig. 3.13(b) demonstrates the dependence of the maximum bond dimension  $D_{\max}$  on the tolerance, suggesting a logarithmic dependence, although the relation displays significant dispersion at low temperature ( $\beta = 1000$ ). The logarithmic scaling means an acceptable increase in computational effort for increasing the accuracy of the QTCI representation.

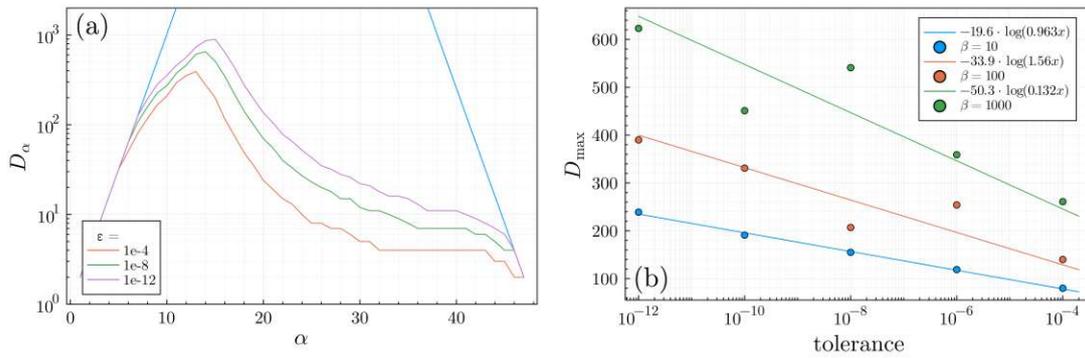


Figure 3.13: (a) QTCI bond dimension for different tolerances. More bond dimensions are required to meet a smaller error. (b) Maximum bond dimension  $D_{\max}$  increases when lowering the tolerance, suggesting a logarithmic dependence.

### 3.2.3 Effect of chemical potential $\mu$

In order to move away from half-filling the chemical potential  $\mu$  was modified

$$\mu \neq U/2 \quad (3.4)$$

Such a chemical potential moves the pole of  $\Sigma$  away from  $\mu$ . The Green's function shows metallic behaviour with a sharp peak in  $\text{Im } G$  and an equally sharp trough in  $\text{Re } G$ , see fig. 3.4 for one dimension. Fig. 3.14 depicts the two-dimensional interacting Green's function  $G$ . The complexity of the structure in  $\mathbf{k}$ -space increases, compared to  $\mu = U/2$  due to the sharper and more pronounced structures. Figs. ??a,d) show  $G$  at half-filling. When increasing  $\mu = U/2 + 2$ , see fig. 3.14(c,e), the troughs deepen and become narrower which is further enhanced when moving to  $\mu = U/2 + 4$ , see fig. 3.14(cf). Therefore, a higher resolution is required to accurately depict the function, which results in a significantly higher demand in computational resources when calculating the QTCI reflected by the increased bond dimensions primarily for higher  $\beta$ , see fig. 3.15. The orange curve shows the bond dimension for half-filling, with a maximum bond dimension of  $D_{\max} = 253$ , whereas the higher values of  $\mu$  are represented by the green ( $\mu = U/2 + 2$ ,  $D_{\max} = 105$ ) and yellow curve ( $\mu = U/2 + 4$ ,  $D_{\max} = 91$ ), respectively.

Summarily, when dealing with calculations off half-filling entanglement

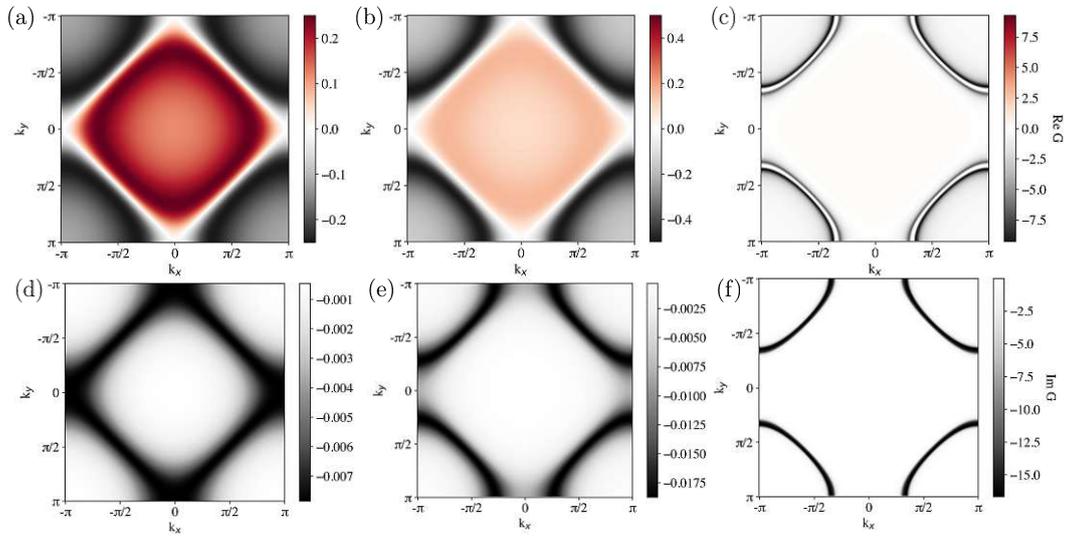


Figure 3.14: Heatmaps of real (top) and imaginary part (bottom) of  $G$  for  $\beta = 100$ ,  $\omega_0$ ,  $\mu = \{U/2, U/2+2, U/2+4\}$ . The structure of the function becomes more complex when moving away from half-filling.

between different length scales gradually increases leading to an increase in bond dimension. This increase is especially drastic at low temperature.

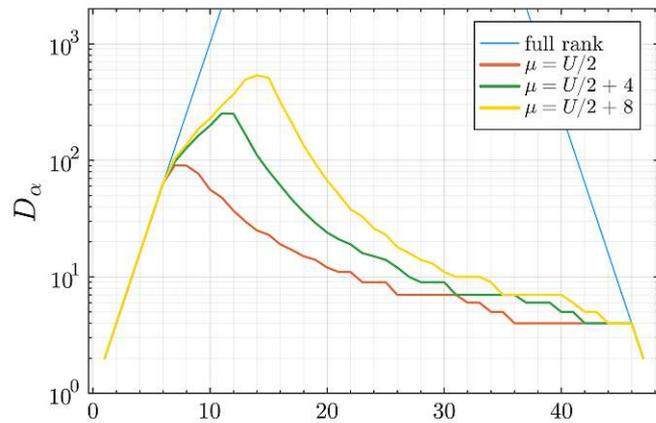


Figure 3.15: Bond dimensions of  $G$  for different chemical potentials  $\mu$ . The increase in bond dimension demonstrates the more complex structure of  $G$  when moving away from half-filling.

### 3.2.4 Fused Quantics Representation

As mentioned in chapter 2.5.2,  $D_{\max}$  can be further reduced by fusing indices of same length scale. This is favourable for certain systems where entanglement is large between fused variables. Figure 3.16(a) compares the bond dimensions in 2D for  $\beta = 100$ ,  $\omega_0$ ,  $R = 24$ ,  $\mu = U/2$ . Fusing two indices  $f(\sigma_{1,1}, \sigma_{2,1}, \sigma_{1,2}, \dots, \sigma_{2,R}) \rightarrow f(\tilde{\sigma}_1, \tilde{\sigma}_2, \dots, \tilde{\sigma}_R)$  proves to have hardly any influence on the bond dimension  $D_\alpha^{\text{fused}} \cong D_\alpha^{\text{interleaved}}$ . Thus, using interleaved or fused indices is interchangeable for this system as it neither decreases the bond dimensions nor is favourable in terms of computation time. On the other hand, when considering a zigzag like behaviour as displayed by  $G_0$  for  $\beta = 100$ ,  $\omega_0$ ,  $R = 24$ ,  $\mu = 0$  fusing sites, smoothes out the curve, minimally reducing bond dimensions, see fig. 3.16(b). The zigzag curve of  $G_0$ , especially noticeable at low temperature  $\beta = \{100, 1000\}$ , suggests that the entanglement between the two variables of same length scale, one corresponding to the  $k_x$ -direction the other one to  $k_y$ , is very high. Thus, every odd bond index tends to have a higher bond dimension than its neighbours. Fusing the indices together, thus, eliminating compression between variables of the same length scale, is expected to smooth the curve. Such behaviour was also found when using SVD on multi-dimensional functions [6].

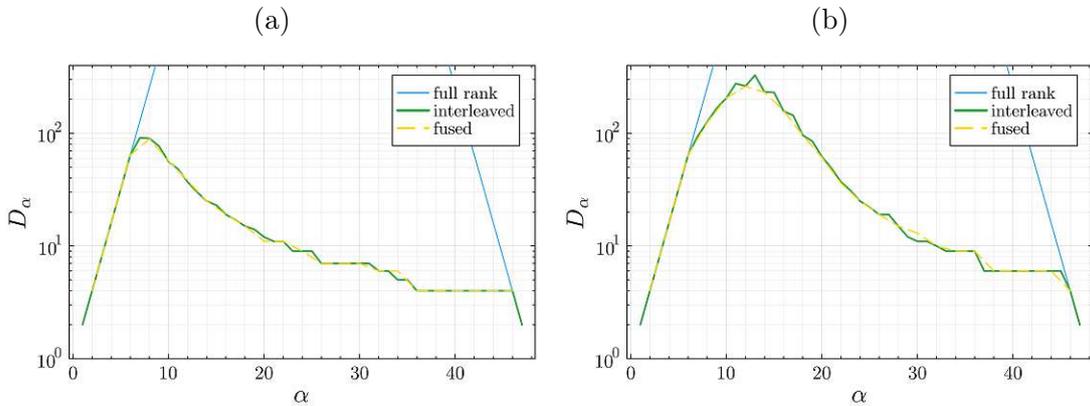


Figure 3.16: Bond dimension for fused sites. For the fused QTCI every second site  $\alpha$  was skipped to make it easily comparable with its interleaved counterpart. (a) Fusing sites does not have an effect for  $G$ , (b) whereas the zigzag curve of  $G_0$  is smoothed out.

### 3.3 Three dimensions

Following the same scheme as for one and two dimensions the expression for the band dispersion for three dimensions is

$$\epsilon(\mathbf{k}) = -2t_x \cos(k_x) - 2t_y \cos(k_y) - 2t_z \cos(k_z) - \mu \quad (3.5)$$

or for a model symmetric in all three directions  $t_x = t_y = t_z = t$ .

$$\epsilon(\mathbf{k}) = -2t \cdot (\cos(k_x) + \cos(k_y) + \cos(k_z)) - \mu \quad (3.6)$$

Calculating in three dimensions considerably increases the computation time and memory cost, making it necessary to switch from a regular Apple M2 processor to the Vienna Science Cluster (VSC). Results were obtained for  $R = 24$ ,  $\epsilon = 10^{-8}$ ,  $\beta = \{1; 10; 100\}$  and  $\omega_0$ . A limit for the maximal bond dimension  $D_{\max}$  was not set. To achieve the same error tolerance,  $D_{\max} \sim 10^3$  considerably increased by a factor of  $\sim 10^2$  ( $\sim 10^3$ ) compared to 2D (1D). As with the other dimensions the effect of temperature on the structure of  $G$  is low at half-filling  $\mu = U/2$ , see fig. 3.17.

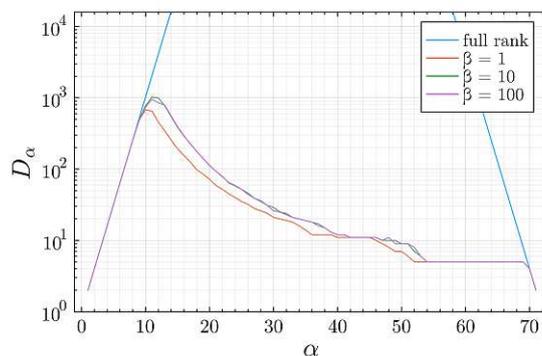


Figure 3.17: Bond dimension of the QTCI of  $G$  for three dimensions with varying temperature at half-filling  $\mu = U/2$ . The cost for decreasing temperature becomes slightly larger in three-dimensional space.

When considering  $G$  outside of half-filling at  $\mu = U/2 + 4$  the QTCI proves to be computationally more challenging increasing  $D_{\max}$  by a few thousand bonds per site, see fig. 3.18. With  $\beta = 1000$  the method runs into its limits with building the QTCI with appropriate grid size ( $R = 24$ ), e.g. taking several days

to finish. This limitation could be avoided by parallelising building the QTCI by *patching* [6]: Instead of deconstructing the whole tensor  $A$ , the  $\mathbf{k}$ -space is partitioned into smaller patches with the motivation that entanglement within a single patch is low and therefore compressible. Although entanglement between different patches is thereby neglected it guarantees parallelisability. Patches can roughly be subdivided into those containing more complex structures within the vicinity of the Fermi surface and these with more uniform data. Consequently,  $D_{\max}$  is individually and parallelly determined saving time and memory. As an additional step, the size of the patches could be evaluated dynamically allowing to increase resolution for structures of interest.

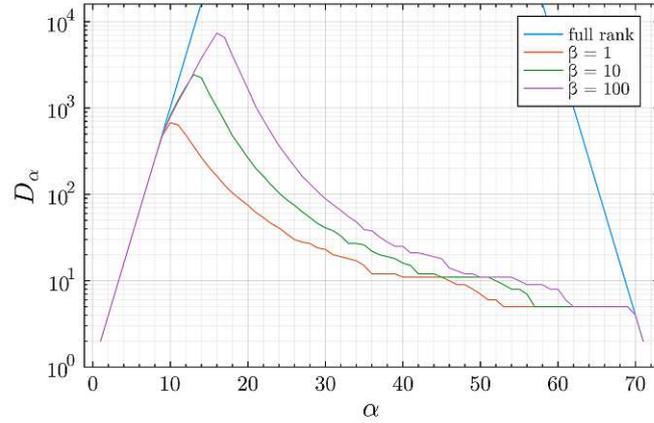


Figure 3.18: Bond dimension of the QTCI of  $G$  for three dimensions with varying temperature outside half-filling at  $\mu = U/2 + 4$ . Decreasing temperature significantly increased the bond dimension.

### 3.3.1 Fused Quantics Representation

When fusing indices of same length scale together for  $n = 3$ :

$$\sigma_\alpha = (\sigma_{\alpha,1}, \sigma_{\alpha,2}, \dots, \sigma_{\alpha,R}) \mid \sigma_{\alpha,i} \in \{0, 1\} \quad (3.7)$$

$$\tilde{\sigma}_\alpha = \sum_{i=1}^3 2^{i-1} \sigma_{\alpha,i} \in \{0, \dots, 7\} \quad (3.8)$$

$$\sigma \rightarrow \tilde{\sigma} = \{\tilde{\sigma}_1, \tilde{\sigma}_2, \dots, \tilde{\sigma}_R\} \quad (3.9)$$

the number of physical indices reduces from  $3R$  to  $R$ , see equ. 2.48 for reference. Comparing results to  $n = 2$ , cf. fig. 3.16 and fig. 3.19, the bond dimensions once again are hardly affected. Most notable is that  $D_{\max}^{\text{fused}} = 1768$  is considerably smaller than  $D_{\max}^{\text{interleaved}} = 2439$  as the variables of the same length scale around the site of  $D_{\max}$  are fused together. Therefore, the entanglement does not show unlike with the interleaved QTCI which produces a larger  $D_{\max}^{\text{interleaved}}$ . Spikes are moderately visible in fig. 3.19 but may also occur for three dimensional data [6].

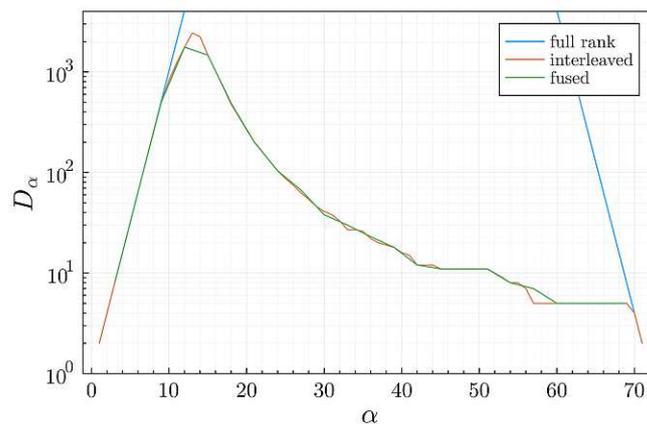


Figure 3.19: Bond dimension in three dimensions with QTCI build from interleaved function (orange) and fused sites (green) for  $\beta = 10$  and  $\mu = 0$ . As with fig. 3.16 fusing sites, smoothes out any zigzag behaviour.

### 3.4 Universal Power Law for $D_{\max}$

Computation time of QTCI strongly depends on the system's bond dimensions which in turn primarily depends on the grid size  $R$  and the maximal bond dimension  $D_{\max}$ .  $R$  can be arbitrarily chosen and  $D_{\max}$  can be estimated from  $\beta$  and the system's dimensions  $n$ . Fig. 3.20 depicts the dependence of  $D_{\max}$  on  $\beta$  outside half-filling at  $\mu = U/2 - 4$ . Fitting the linear dependence for a log-log plot with a power law the universal relation

$$D_{\max}(\beta, n) = A_n \cdot \beta^{(n-1)B} \quad (3.10)$$

is derived, with the factor  $A_n$  depending on  $n$  and a constant  $B \cong 0.253 \pm 0.00623$ . As  $D_{\max}$  quantifies the entanglement across various length scales, this observation may inspire further exploration into the underlying reasons for the power law and prompt investigations into entangled systems extending beyond one dimension. Among others, (functional) renormalisation groups [(f)RG] might provide a useful approach, as RG, similarly to QTCI, inherently deals with the separation of length scales, as demonstrated for the two-dimensional Ising-model by Wilson [8].

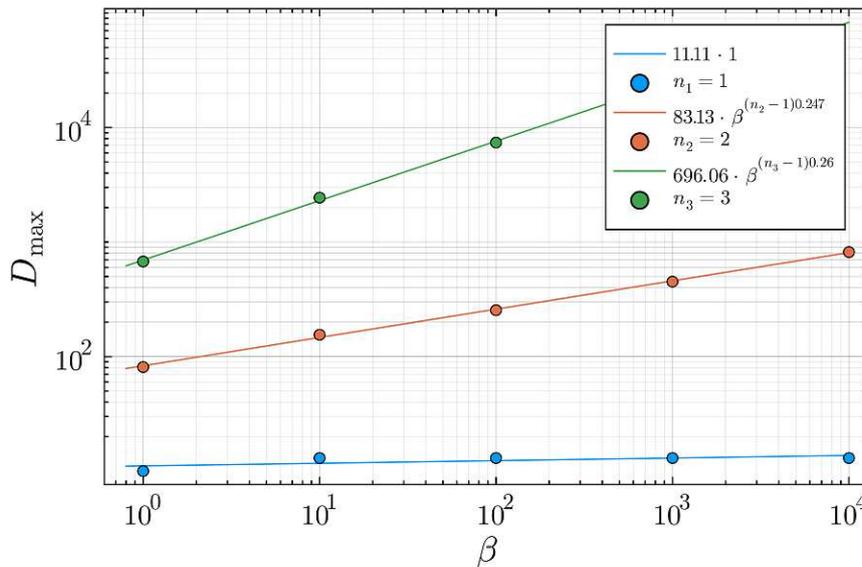


Figure 3.20: Maximum bond dimensions  $D_{\max}$  in dependence of the inverse temperature  $\beta$ . Fitting a power law demonstrates a universal relation between these two parameters for different dimensions  $n \in \{1, 2, 3\}$ .

## 4 Conclusion and Outlook

Applying QTCI to build a MPS of the interacting Green's function  $G$  in  $\mathbf{k}$ -space establishes that the method is capable of efficiently compressing and simplifying multi-dimensional tensors. Thus, QTCI provides an alternative route to the well established SVD. Using a quantics representation for the  $k$ -values allows the separation of different length scales. Assuming that different length scales are minimally entangled, this reduces the required bond dimensions and cost of building the MPS. As with the skeleton representation, QTCI does not require the complete tensor to build the MPS, in contrast to SVD. The algorithm searches iteratively for a new pivot, adding a new set of a row and a column to the pivot matrix when the error is maximal. This generates a quasi-optimal result, but is less time and memory consuming than calculating the SVD. QTCI is especially efficient when applied to repetitive data but loses its advantages when confronted with randomised tensors, as such structures are not compressible.

The algorithm tunes the accuracy of the MPS representation to a given the tolerance  $\epsilon$ . Although not directly corresponding to the error of the QTCI compared to the original function, it is an adequate and efficient indicator of the overall error. Bond dimension and, thus, computational cost, increase, when demanding a higher accuracy. Nevertheless, correlation between  $\epsilon$  and the maximum bond dimension  $D_{\max}$  suggests a logarithmic dependency and, therefore, a moderate increase.

The bond dimension  $D_\alpha$  per site  $\alpha$  generally reflects the complexity and intricacy of a function. This is seen when going to low temperature ( $\beta = 1000$ ) or moving away from half-filling in the Hubbard-model ( $\mu \neq 0$ ). Dealing with the higher-dimensional  $\mathbf{k}$ -space proved to be particularly challenging, as  $D_{\max}$  for  $\beta = 100$  increased from  $\mathcal{O}(10^1)$  for 1D to  $\mathcal{O}(10^2)$  for 2D to  $\mathcal{O}(10^3)$  for 3D. Additionally, the grid size scales with  $2^{nR}$ , thereby, increasing the number of sites by  $R$  for each

dimension added.

Fusing variables of same length scale together reduces the number of sites to  $R$ , each index now taking values from  $[0, 2^n - 1]$ . In general fusing sites has no significant advantage in terms of computation cost or compressibility, except when the bond dimension exhibit a zigzag behaviour, meaning that entanglement between the same length scale is larger than between different length scales. For this case fusing variables smoothes out the curve of the bond dimensions.

Lastly, a universal power law for the dependence of  $D_{\max}$  on  $\beta$  and the system's dimension  $n$  was found, stating that  $D_{\max} = A_n \cdot \beta^{(n-1)0.253}$ . This provides an estimation for computational cost based on just a few system parameters. The mathematical derivation of this power law remains unclear, but efforts motivated by renormalisation group theory to explain entanglement of systems beyond one dimension might be applicable to this problem.

While it was established that QTCI is an efficient and reliable tool for dealing with large amounts of data, it proved to be very time consuming for three-dimensional  $k$ -space. This issue can be resolved, by parallelising the code and implementing dynamic patching. This requires to split a large tensor into more manageable sections. For each section the QTCI is calculated separately. It thereby neglects entanglement between patches in favour of parallelisation.

Additionally, revising the selection process of new pivots may further reduce the time requirement to build the MPS. As for now every (available) column and row is considered and pivots are selected for producing the maximal error. Considering only a reduced sample will find a less optimal compression, but simultaneously reduce the computational cost and time for adding a new pivot. This approach naturally requires a suitable sampling algorithm.

As a next step, QTCI may be applied to other problems in many-body physics. The (functional) renormalisation groups inertly play into the concept of length scale separation, as does QTCI. The two methods together seem to be a promising combination.

Furthermore, studying critical phenomena with QTCI might be particularly challenging. Naturally, the correlation length diverges when a system approaches a critical point, opposing the basic concept of QTCI, stating that entanglement is small and largely compressible. Investigating how QTCI performs when confronted

with critical phenomena is an intriguing research topic for the future.

Lastly, QTCI is a very powerful tool for numerical integration. First compressing complex structures, before calculating an integral, will prove an efficient alternative to other common methods, like Monte-Carlo integration.



# Bibliography

- [1] M. P. Marder, *Condensed Matter Physics* (Wiley, 2010).
- [2] A. Zagoskin, *Quantum theory of many-body systems: techniques and applications*, Graduate Texts in Physics (Springer International Publishing, 2016).
- [3] M. Wallerberger, H. Shinaoka, and A. Kauch, “Solving the bethe-salpeter equation with exponential convergence”, *Phys. Rev. Res.* **3** (2021).
- [4] U. Schollwoeck, “The density-matrix renormalization group in the age of matrix product states”, *Annals of Physics* **326** (2010).
- [5] M. K. Ritter, Y. Núñez Fernández, M. Wallerberger, J. von Delft, H. Shinaoka, and X. Waintal, “Quantics tensor cross interpolation for high-resolution parsimonious representations of multivariate functions”, *Phys. Rev. Lett.* **132** (2024).
- [6] H. Shinaoka, M. Wallerberger, Y. Murakami, K. Nogaki, R. Sakurai, P. Werner, and A. Kauch, “Multiscale space-time ansatz for correlation functions of quantum systems based on quantics tensor trains”, *Phys. Rev. X* **13** (2023).
- [7] S. A. Goreinov, E. E. Tyrtysnikov, and N. L. Zamarashkin, “A theory of pseudoskeleton approximations”, *Linear Algebra and its Applications* **261** (1997).
- [8] K. G. Wilson, “The renormalization group: critical phenomena and the kondo problem”, *Rev. Mod. Phys.* **47** (1975).
- [9] L. J. P. Ament, M. van Veenendaal, T. P. Devereaux, J. P. Hill, and J. van den Brink, “Resonant inelastic x-ray scattering studies of elementary excitations”, *Rev. Mod. Phys.* **83** (2011).

- [10] H. Schober, “An introduction to the theory of nuclear neutron scattering in condensed matter”, *Journal of Neutron Research* **17** (2014).
- [11] A. Fetter and J. Walecka, *Quantum theory of many-particle systems*, Dover Books on Physics (Dover Publications, 2012).
- [12] J. M. Luttinger, “Analytic properties of single-particle propagators for many-fermion systems”, *Phys. Rev.* **121** (1961).
- [13] G. Rohringer, A. Valli, and A. Toschi, “Local electronic correlation at the two-particle level”, *Phys. Rev. B* **86** (2012).
- [14] E. Howard, I. S. Chowdhury, and I. Nagle, *Matrix Product State Representations for Machine Learning* (Springer International Publishing, 2021).
- [15] Y. Núñez Fernández, M. Jeannin, P. T. Dumitrescu, T. Kloss, J. Kaye, O. Parcollet, and X. Waintal, “Learning feynman diagrams with tensor trains”, *Phys. Rev. X* **12** (2022).
- [16] S. Dolgov and D. Savostyanov, “Parallel cross interpolation for high-precision calculation of high-dimensional integrals”, *Computer Physics Communications* **246** (2020).
- [17] M. Born and R. Oppenheimer, “Zur Quantentheorie der Molekeln”, *Annalen der Physik* **389** (1927).
- [18] S. A. Goreinov and E. E. Tyrtyshnikov, “Quasioptimality of skeleton approximation of a matrix in the Chebyshev norm”, *Doklady Mathematics* **83** (2011).
- [19] H. Cheng, Z. Gimbutas, P. G. Martinsson, and V. Rokhlin, “On the Compression of Low Rank Matrices”, *SIAM Journal on Scientific Computing* **26** (2005).
- [20] D. V. Savostyanov, “Quasioptimality of maximum-volume cross interpolation of tensors”, *Linear Algebra and its Applications* **458** (2014).
- [21] B. Settles, *Active learning literature survey*, Computer Sciences Technical Report 1648 (University of Wisconsin–Madison, 2009).
- [22] B. N. Khoromskij, “ $O(d \log N)$ -Quantics Approximation of  $N$ -d Tensors in High-Dimensional Numerical Modeling”, *Constructive Approximation* **34** (2011).

- [23] University of Vienna, *Supercomputer – VSC*, <https://zid.univie.ac.at/vsc/> (visited on 01/10/2024).
- [24] N. Wagner, L. Crippa, A. Amaricci, P. Hansmann, M. Klett, E. J. König, T. Schäfer, D. D. Sante, J. Cano, A. J. Millis, A. Georges, and G. Sangiovanni, “Mott insulators with boundary zeros”, *Nature Communications* **14** (2023).