ParquetIR.jl

Efficiently solving the parquet equations using a sparse representation

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Technische Physik

eingereicht von

Samuel Badr, B.Sc.
Matrikelnummer 01527657

an der Fakultät für Physik

der Technischen Universität Wien

Betreuung: Prof. Karsten Held
Mitwirkung: Dr. Anna Kauch
Dr. Markus Wallerberger

Wien, 22. Oktober 2024

Samuel Badr | Karsten Held

Kurzfassung

In dieser Arbeit präsentieren und diskutieren wir den ersten Solver für die Parquet-Gleichungen (den wir `ParquetIR.jl` nennen), der auf der kürzlich entwickelten intermediate representation (IR) Basis basiert und in der Programmiersprache Julia geschrieben ist. Zwei-Teilchen-Größen werden in einer sparsamen Darstellung gespeichert, was eine erhebliche Reduktion der Speicher- und Rechenzeitanforderungen ermöglicht. Der Solver verwendet als Eingabe (eine Näherung der) irreduziblen Wechselwirkungsfunktion und die nicht-wechselwirkende Einteilchen-Green-Funktion. Anschließend wird ein hochmoderner Fixpunkt-Solver eingesetzt, um eine numerische Lösung der Parquet-Gleichungen zu finden. Dies liefert die volle Wechselwirkungsfunktion, die Einteilchen-Green-Funktion und die Selbstenergie. Wir diskutieren den theoretischen Hintergrund der Parquet-Gleichungen, die Implementierungsdetails des Solvers und wenden ihn auf das Hubbard-Atom sowie auf das 4×4 Hubbard-Modell auf einem quadratischen Gitter an. Dabei zeigen wir Vergleiche mit einem traditionellen Ansatz. Die Ergebnisse stimmen gut mit den Referenzdaten überein und demonstrieren das Potenzial von `ParquetIR.jl`, größere Systeme zu behandeln.

Abstract

In this thesis, we present and discuss the first solver for the parquet equations (which we call `ParquetIR.jl`) based on the recently developed intermediate representation (IR) basis and written in the Julia programming language. Two-particle quantities are stored in a sparse representation enabling a significant reduction in memory and computation time requirements. The solver takes as input (an approximation of) the irreducible vertex and the non-interacting one-particle Green's function. It then employs a state-of-the-art fixed-point solver to find a numerical solution to the parquet equations. This yields the full vertex, the one-particle Green's function, and the self-energy. We discuss the theoretical background of the parquet equations and the implementation details of the solver and apply it to the Hubbard atom and the 4×4 Hubbard model on a square lattice; and show comparisons with a traditional approach. The results exhibit good agreement with the benchmark data and demonstrate the potential of `ParquetIR.jl` to tackle larger systems.

Contents

CHAPTER 1

Introduction

In condensed matter physics, an area of great interest is the electronic behavior of solids. The relevant systems are ones comprising many interacting quantum particles, which makes them hard to grasp. Exciting phenomena like strong response to external perturbations, giant magnetoresistance, metal-to-insulator transitions and superconductivity are all emergent properties of these systems. Fortunately, a wealth of theory and techniques has been developed to tackle the problems inherent in large quantum systems. Our specific focus will lie on two-particle vertices, which are crucial for – among other things – scattering processes, which in turn are crucial for the understanding of transport properties in solids, like electrical conductivity.

The fundamental object of interest in this regard will be the generalization of the one-particle propagator, the two-particle Green's function. While one-particle response functions are routinely calculated using techniques like the dynamical mean-field theory (DMFT) [1], the two-particle Green's function is a much more challenging object. The time and memory required to compute it scale poorly with inverse temperature $\beta = 1/T$ and number of orbitals considered. To remedy this, static approximations with only the zero frequency component [2, 3, 4] or reduced frequency dependence [5, 6, 7, 8, 9, 10] are often used. Fully dynamical calculations (usually two-particle extensions of DMFT [11, 12, 13, 14, 15, 16, 17]) however quickly hit a wall, and are to date only possible at high temperature and for a few orbitals. Additionally, two-particle calculations tend to boast a high dynamic range, i.e. feature energies ranging over many orders of magnitude [18]. This already rules out many approximate methods. With many interesting parameter regions thus inaccessible, a solution is desirable and only possible through a paradigm shift, e.g. by using domain-specific modeling to compress data.

The basic formalism we choose is the elegant parquet method [19, 20], devised already in the 1960s. It revolves around solving a set of diagrammatic equations, the so-called parquet equations, named for the decorative style of flooring. They are a set of exact relations between one- and two-particle quantities that arise from a couple of

diagrammatic-topological arguments. Since their introduction, they have been standing as a formidable challenge, requiring extensive computing resources even at small resolutions, prohibiting potentially interesting applications.

In the present work, we develop an entirely new parquet solver called `ParquetIR.jl` in the Julia programming language [21]. We apply the solver to a test case of the Hubbard atom, where the exact solution is known [22] and to the 4×4 Hubbard model on a square lattice as first test examples.

1.1 Outline

We begin in chapter 2 by briefly defining the physical model – the Hubbard model – that we will endeavor to solve before introducing the parquet equations, closely following our main reference, Julian Mangott's excellent 2022 master's thesis [23]. Here, we will only roughly sketch out the derivations and refer the reader to the original work for details about the calculations.

In the main part, chapter 3, we describe the implementation of the aforementioned equations. In this context, we showcase the two main aspects that are original to our software. Firstly, we directly exploit the problem's fixed-point structure, applying a state-of-the-art fixed-point solver to the parquet equations, improving convergence. Secondly, we use a sparse representation of the two-particle Green's function developed by Hiroshi Shinaoka and Markus Wallerberger [24], which allows us to compress the data and reduce the calculational requirements.

In chapter 4, we present the output of our solver, including performance scaling analysis and comparison to a benchmark reference implementation. Finally, we summarize our results and give an outlook on possible future research directions in chapter 5.

CHAPTER 2

Definitions

2.1 Model

2.1.1 Hubbard model

The model we will consider in the following is the Hubbard model [25], an approximate model for electrons in a solid. The solid is approximated as consisting of a grid of sites – corresponding to orbitals around atoms – each of which provides "space" for up to two electrons, one spin-up and one spin-down. In addition to having a kinetic energy, each electron is subject to the Coulomb force of strength U_i, pushing it away from its current site i in case there is a second electron at the site. This idea is formalized in the Hamiltonian

$$H = -t \sum_{\langle ij \rangle, \sigma} \left(c^\dagger_{i,\sigma} c_{j,\sigma} + \text{h.c.} \right) + U \sum_i n_{i\uparrow} n_{i\downarrow} \tag{2.1}$$

where $n_{i\sigma} = c^\dagger_{i\sigma} c_{i\sigma}$ is the spin-density operator for spin σ on site i.

The first sum is to be performed over all pairs $\langle ij \rangle$ of sites i and j as well as both spin orientations $\sigma \in \{\uparrow, \downarrow\}$, and the second sum is to be performed over all sites i. The first term, describing the hopping from one site to another, is parametrized by the hopping integral t.

The second term is the Coulomb interaction between two electrons at the same site, i.e. any site with both "electron slots" filled adds U_i to the energy.

We will also assume time-translation symmetry as well as SU(2) symmetry in our equations.

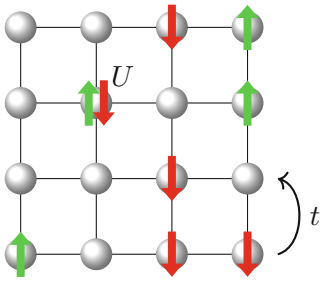

Figure 2.1: The 2-dimensional Hubbard model on a square lattice.

2.1.2 Green's functions

A useful tool in the description of many-electron systems is the system's *one-particle Green's function* (or propagator) $G(12)$ which describes the probability amplitude of a particle propagating from state (i_2, σ_2) at imaginary time τ_2 to state (i_1, σ_1) at imaginary time τ_1

$$G(12) = -\left\langle \mathcal{T} c(1) c^\dagger(2) \right\rangle . \tag{2.2}$$

Here we adopt the notation from [23], wherein tuples $(i_\alpha, \sigma_\alpha, \tau_\alpha)$ of site, spin and imaginary time are represented by a single index $\alpha \in \mathbb{Z}$, using the definition

$$c(\alpha) = \mathrm{e}^{\tau_\alpha H} c_{i_\alpha, \sigma_\alpha} \mathrm{e}^{-\tau_\alpha H}. \tag{2.3}$$

An object of pivotal importance for our work is a generalization of this, the *two-particle Green's function*

$$G(1234) = \left\langle \mathcal{T} c(1) c^\dagger(2) c(3) c^\dagger(4) \right\rangle . \tag{2.4}$$

It is the probability amplitude for two electrons starting in the (loosely speaking) states 2 and 4 and ending up in states 1 and 3.

In terms of Feynman diagrams, the two-particle Green's function is the sum of all diagrams with four external lines. If we take only the connected diagrams and truncate the external lines, we obtain the so-called *full two-particle vertex* F. This is a key object, because we will be able to write down a number of diagrammatic equations in order to compute the Green's functions, our ultimate objective.

2.1.3 Self-energy

With the non-interacting Hamiltonian H_0[1], we define the *non-interacting Green's function* by

$$G_0(12) = -\left\langle \mathcal{T} c(1) c^\dagger(2) \right\rangle_0 , \tag{2.5}$$

where $\langle \cdot \rangle_0 \equiv \frac{1}{Z_0} \mathrm{tr}(\mathrm{e}^{-\beta H_0} \cdot)$ and $Z_0 = \mathrm{tr}(\mathrm{e}^{-\beta H_0})$ is the partition function.

Together with the bare interaction vertex

$$U(1234) = U \sum_{ijkl} \delta(\tau_1 - \tau_2)\delta(\tau_2 - \tau_3)\delta(\tau_3 - \tau_4)\delta_{i_1 i}\delta_{i_2 k}\delta_{i_3 j}\delta_{i_4 l} \tag{2.6}$$

the non-interacting Green's function forms the basic building blocks from which all other objects in our diagrammatic formalism are assembled, c.f. fig. 2.2. The full Green's function then is the sum of all connected diagrams. These constituent diagrams may be classified according to whether they are 1-particle irreducible (1PI). Here a diagram is said to be 1PI if there is no internal line whose removal would result in two disconnected

[1] In our case, $H_0 = -t \sum_{\langle ij \rangle, \sigma} \left(c^\dagger_{i,\sigma} c_{j,\sigma} + \text{h.c.} \right)$.

diagram parts. The sum of all 1PI diagrams with both external lines amputated is called *self-energy* Σ.

If we now reconstruct the full Green's function from these parts, we get the *Dyson equation*

$$G(12) = G_0(12) + G_0(13)\Sigma(34)G(42), \tag{2.7}$$

where summation over the repeated variables (3 and 4) is implied. Multiplying from the left by G_0^{-1} and from the right by G^{-1} yields the alternative form (which is the one we will use in our implementation later)

$$G = \left(G_0^{-1} - \Sigma\right)^{-1}. \tag{2.8}$$

2.2 Parquet equations

The term *parquet equations* is used to denote a set of exact equations that are simultaneously satisfied by the two-particle vertex F, the one-particle Green's function G and the three additionally introduced irreducible vertices in one scattering channel (particle-hole Γ^{ph}, transversal particle-hole $\Gamma^{\overline{\mathrm{ph}}}$, and particle-particle Γ^{pp}) and the fully irreducible (not reducible in any of the channels) vertex Λ (for a detailed introduction see e.g. [26]). The parquet equations are an unbiased generalization of the Dyson equation to the two-particle case and their representation in terms of Feynman diagrams is shown in fig. 2.2. The two particle Green's function can be represented as a function of Matsubara frequencies and momenta via the Fourier transform

$$G^{ijlm}_{\sigma_1\ldots\sigma_4}(\tau_1,\tau_2,\tau_3,\tau_4) = \sum_{k_1,k_2,k_3,k_4} G^{k_1k_2k_3k_4}_{\sigma_1\ldots\sigma_4} \mathrm{e}^{-\mathrm{i}\boldsymbol{k}_1\boldsymbol{r}_i - \mathrm{i}\boldsymbol{k}_2\boldsymbol{r}_j - \mathrm{i}\boldsymbol{k}_3\boldsymbol{r}_l - \mathrm{i}\boldsymbol{k}_4\boldsymbol{r}_m} \mathrm{e}^{-\mathrm{i}\nu_1\tau_1 - \mathrm{i}\nu_2\tau_2 - \mathrm{i}\nu_3\tau_3 - \mathrm{i}\nu_4\tau_4} \tag{2.9}$$

where we use a compact combined momentum and frequency notation $k = (\boldsymbol{k}, \nu)$ and also include all $1/(\beta N)$ prefactors connected with momentum and frequency in the definition of the $\sum$ symbol, i.e. $\sum_k := \frac{1}{\beta N}\sum_{\boldsymbol{k},\nu}$. Momentum and energy conservation means that only three of the four momentum-frequency arguments are independent. Taking energy and momentum conservation into account, usually one parametrizes the four arguments by three momenta $(\boldsymbol{k}, \boldsymbol{k}', \boldsymbol{q})$ and frequencies (ν, ν', ω), the first two being fermionic and the third bosonic.

In the following we assume SU(2) symmetry which means that only two spin combinations need to be considered: $\sigma_1\sigma_2\sigma_3\sigma_4 = \uparrow\uparrow\downarrow\downarrow$ which we will call $\uparrow\downarrow$ for short following [23] and $\uparrow\uparrow\uparrow\uparrow$, abbreviated $\uparrow\uparrow$. Another abbreviation we will need is $\overline{\uparrow\downarrow}$ which stands for $\uparrow\downarrow\downarrow\uparrow$.

The full two-particle vertex $F^{kk'q}_{\sigma\sigma'}$, diagrammatically the sum of all connected two-particle diagrams with external lines amputated, is then defined in the so-called particle-hole notation [27] (we will later also introduce the particle-particle notation) through

$$G^{kk'q}_{\sigma\sigma'} = G^k G^{k'} \delta_{q0} - G^k G^{k+q} \delta_{kk'} \delta_{\sigma\sigma'} - G^k G^{k+q} F^{kk'q}_{\sigma\sigma'} G^{k'} G^{k'+q}, \tag{2.10}$$

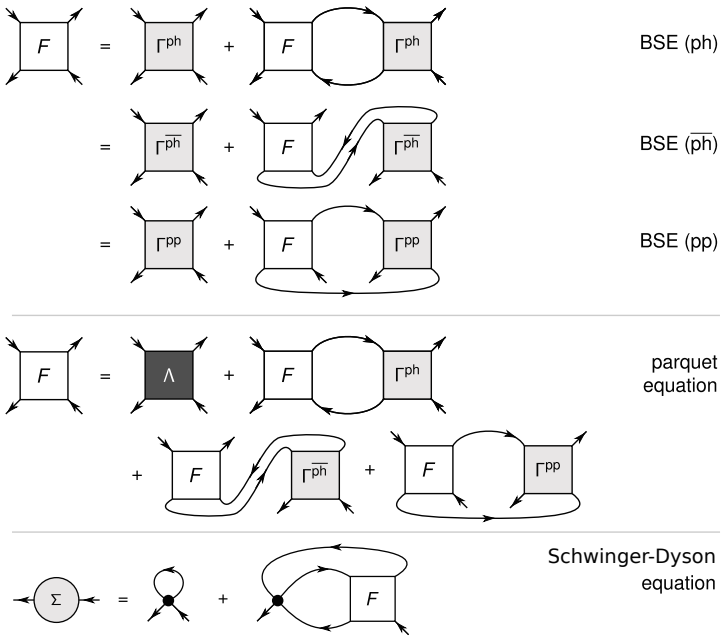

Figure 2.2: Diagrams illustrating the parquet equations. Lines correspond to one-particle Green's functions G and bold dots correspond to bare-interaction vertices U.

where for the one-particle Green's function G^k we dropped the spin index, since in the SU(2) symmetric case it is spin-diagonal and equal for both spins. It is useful to use so-called spin-diagonal notation for vertices. For vertices in the particle-hole notation we define the following linear combinations:

$$X_\mathrm{d} \coloneqq X^\mathrm{ph}_{\uparrow\uparrow} + X^\mathrm{ph}_{\uparrow\downarrow} \tag{2.11a}$$

$$X_\mathrm{m} \coloneqq X^\mathrm{ph}_{\uparrow\uparrow} - X^\mathrm{ph}_{\uparrow\downarrow} \tag{2.11b}$$

with X being either F, the irreducible vertex Γ or Λ, or the reducible vertex Φ (which we will define in the following).

The full set of parquet equations includes:

- Bethe-Salpeter equation (BSE) in the particle-hole (ph) channel (the BSE in the particle-hole transverse ($\overline{\mathrm{ph}}$) channel is related through a transformation of the momenta and frequencies)
- BSE in the particle-particle (pp) channel
- Parquet equation adding contributions from all scattering channels
- Schwinger-Dyson equation (SDE), relating the self-energy Σ to the full vertex F
- Dyson equation yielding the full one-particle Green's function from the self-energy

2.2.1 Bethe-Salpeter equation

A propagator-reducible diagram is one that may be cut into two diagrams by cutting two internal one-particle Green's function lines. Depending on which external lines remain connected, we count the diagram as belonging to one of three channels:

- ph (particle-hole)
- $\overline{\text{ph}}$ (particle-hole transverse)
- pp (particle-particle)

For each channel $r \in \{\text{ph}, \overline{\text{ph}}, \text{pp}\}$, the sum of all diagrams reducible in that channel shall be called Φ^r; the sum of the irreducible diagrams is called Γ^r. In this way we obtain a decomposition of the full vertex into these two classes

$$F = \Gamma^r + \Phi^r. \tag{2.12}$$

Furthermore, every reducible diagram can be written as the connection of an irreducible diagram with the full vertex:

$$\Phi^{\text{ph}}(1234) = \quad \Gamma^{\text{ph}}(1256)G(67)G(85)F(7834) \tag{2.13a}$$
$$\Phi^{\overline{\text{ph}}}(1234) = -\Gamma^{\overline{\text{ph}}}(1654)G(67)G(85)F(7238) \tag{2.13b}$$
$$\Phi^{\text{pp}}(1234) = \tfrac{1}{2}\,\Gamma^{\text{pp}}(1536)G(67)G(58)F(7284) \tag{2.13c}$$

Where again repeated variables, i.e. 5, 6, 7 and 8, are implicitly summed over as in eq. (2.7).

So, taking these facts together, in the particle-hole channel we can write the *Bethe-Salpeter equation* (BSE)

$$F^{kk'q}_{\text{d/m}} = \Gamma^{kk'q}_{\text{d/m}} + \sum_{k_1} F^{kk_1q}_{\text{d/m}} G^{k_1} G^{k_1+q}\, \Gamma^{k_1k'q}_{\text{d/m}} \tag{2.14}$$

The ph channel corresponds to the first line of diagrams in fig. 2.2 and to eq. (2.14).

The other two channels – $\overline{\text{ph}}$ and pp – are similarly displayed in the figure. Analogously to eq. (2.11a) we define a *singlet* and a *triplet* channel for particle-particle diagrams

$$X_{\text{s}} \coloneqq X^{\text{pp}}_{\uparrow\downarrow} + X^{\text{pp}}_{\overline{\uparrow\downarrow}} \tag{2.15a}$$
$$X_{\text{t}} \coloneqq X^{\text{pp}}_{\uparrow\downarrow} - X^{\text{pp}}_{\overline{\uparrow\downarrow}} \tag{2.15b}$$

that give rise to 4 additional BSEs, 1 in the singlet and 3 in the triplet channel (so these names are not all that creative), but the 3 triplet BSEs are degenerate.

The BSE equations in the particle-hole and transversal particle-hole channels are not independent and $\Gamma^{\overline{\text{ph}}}$ can be obtained from Γ^{ph} using the crossing symmetry [27]. This leaves only four BSEs that need to be accounted for.

2.2.2 Parquet equation

In the previous subsection we defined three kinds of reducible vertices, depending on which corners of the vertices stay connected by the one-particle Green's functions when cutting two Green's function lines. One can show that a vertex reducible in one channel is irreducible in the two other channels [20, 26]. This allows for a unique classification of all diagrams contained in F with respect to their two-particle reducibility: A given diagram is either reducible in one of the three channels (contained in Φ^{ph}, $\Phi^{\overline{\mathrm{ph}}}$, or Φ^{pp}) or fully irreducible. The sum of these fully irreducible diagrams is called the fully irreducible vertex Λ. The full vertex is thus given by the parquet equation

$$F = \Lambda + \Phi^{\mathrm{ph}} + \Phi^{\overline{\mathrm{ph}}} + \Phi^{\mathrm{pp}}. \tag{2.16}$$

In order to get concrete relations between the respective spin components, as well as to make use of crossing symmetry, we need to rewrite the above parquet equation in d/m and s/t components.

We also introduce "channel-native" vertex conventions [23] by

$$F(1234) = F^{\mathrm{ph}}(12 \mid 34) = F^{\overline{\mathrm{ph}}}(14 \mid 32) = F^{\mathrm{pp}}(13 \mid 24) \tag{2.17}$$

which will allow us to unify the Bethe-Salpeter equations from eq. (2.13) to

$$\Phi^r(12 \mid 34) = \Gamma^r(12 \mid 56) X_0^r(56 \mid 78) F^r(78 \mid 34) \tag{2.18}$$

with channel-wise bare susceptibilities

$$X_0^{\mathrm{ph}}(12 \mid 34) = G(23)G(41) \tag{2.19a}$$
$$X_0^{\overline{\mathrm{ph}}}(12 \mid 34) = -G(23)G(41) \tag{2.19b}$$
$$X_0^{\mathrm{pp}}(12 \mid 34) = \tfrac{1}{2} G(23)G(14). \tag{2.19c}$$

This way we obtain channel-wise parquet-equations

$$F^{\mathrm{ph}}(12 \mid 34) = \Lambda^{\mathrm{ph}}(12 \mid 34) + \Phi^{\mathrm{ph}}(12 \mid 34) + \Phi^{\overline{\mathrm{ph}}}(14 \mid 32) + \Phi^{\mathrm{pp}}(13 \mid 24) \tag{2.20a}$$
$$F^{\overline{\mathrm{ph}}}(12 \mid 34) = \Lambda^{\overline{\mathrm{ph}}}(12 \mid 34) + \Phi^{\mathrm{ph}}(14 \mid 32) + \Phi^{\overline{\mathrm{ph}}}(12 \mid 34) + \Phi^{\mathrm{pp}}(13 \mid 42) \tag{2.20b}$$
$$F^{\mathrm{pp}}(12 \mid 34) = \Lambda^{\mathrm{pp}}(12 \mid 34) + \Phi^{\mathrm{ph}}(13 \mid 24) + \Phi^{\overline{\mathrm{ph}}}(14 \mid 23) + \Phi^{\mathrm{pp}}(12 \mid 34) \tag{2.20c}$$

with frequency permutations.

Taking into account the symmetries (time-translation and SU(2)) we get [23]

$$F_{\mathrm{d}}^{kk'q} = \Lambda_{\mathrm{d}}^{kk'q} + \Phi_{\mathrm{d}}^{kk'q} - \tfrac{1}{2}\Phi_{\mathrm{d}}^{k(k+q)(k'-k)} - \tfrac{3}{2}\Phi_{\mathrm{m}}^{k(k+q)(k'-k)} + \tfrac{3}{2}\Phi_{\mathrm{t}}^{kk'(q+k+k')} + \tfrac{1}{2}\Phi_{\mathrm{s}}^{kk'(q+k+k')} \tag{2.21a}$$

$$F_{\mathrm{m}}^{kk'q} = \Lambda_{\mathrm{m}}^{kk'q} + \Phi_{\mathrm{m}}^{kk'q} - \tfrac{1}{2}\Phi_{\mathrm{d}}^{k(k+q)(k'-k)} + \tfrac{1}{2}\Phi_{\mathrm{m}}^{k(k+q)(k'-k)} + \tfrac{1}{2}\Phi_{\mathrm{t}}^{kk'(q+k+k')} - \tfrac{1}{2}\Phi_{\mathrm{s}}^{kk'(q+k+k')} \tag{2.21b}$$

$$F_{\mathrm{s}}^{kk'q} = \Lambda_{\mathrm{s}}^{kk'q} + \Phi_{\mathrm{s}}^{kk'q} + \tfrac{1}{2}\Phi_{\mathrm{d}}^{kk'(q-k-k')} - \tfrac{3}{2}\Phi_{\mathrm{m}}^{kk'(q-k-k')} + \tfrac{1}{2}\Phi_{\mathrm{d}}^{k(q-k')(k'-k)} - \tfrac{3}{2}\Phi_{\mathrm{m}}^{k(q-k')(k'-k)} \tag{2.21c}$$

$$F_{\mathrm{t}}^{kk'q} = \Lambda_{\mathrm{t}}^{kk'q} + \Phi_{\mathrm{t}}^{kk'q} + \tfrac{1}{2}\Phi_{\mathrm{d}}^{kk'(q-k-k')} + \tfrac{1}{2}\Phi_{\mathrm{m}}^{kk'(q-k-k')} - \tfrac{1}{2}\Phi_{\mathrm{d}}^{k(q-k')(k'-k)} - \tfrac{1}{2}\Phi_{\mathrm{m}}^{k(q-k')(k'-k)} \tag{2.21d}$$

The shifted indices seen here are a consequence of us representing the vertices in channel-native conventions. This enables the Bethe-Salpeter equation to have the same form in all channels but makes it necessary to convert between the conventions in the parquet equation.

2.2.3 Schwinger-Dyson equation

Finally, the *Schwinger-Dyson equation* (SDE) gives us a way of computing the self-energy Σ from the full vertex F and the one-particle Green's function G (c.f. fig. 2.2)

$$\Sigma^k = -\int_{k'} UG(k')\mathrm{e}^{\mathrm{i}k'0^-} - \tfrac{1}{2}\iint_{k'q} UG(k+q)G(k')G(k'+q)F^{\mathrm{ph},k'kq}. \tag{2.22}$$

Finally, the Dyson equation was already given in eq. (2.7)

2.2.4 Convergence and Uniqueness

With the so-called fully irreducible vertex Λ as input and some initial guess for the Γ's these equations can be iteratively solved to get the solution of a many-body problem (e.g. the Hubbard model). It is tempting to think that with an exactly known irreducible vertex Λ one could always reach the correct solution in this way, provided the chosen iterative scheme converges. However, depending on the model being solved there exist multiple solutions of the parquet equations. For example, in the Hubbard atom (i.e. the Hubbard model with $t = 0$) around and above the first so-called divergence line at $\beta U \approx 3.6$, the parquet method converges to a non-physical solution. It stands to reason that this phenomenon is not limited to the Hubbard atom, but for the parameters we tested the Hubbard model does not exhibit it. Ameliorating this with more sophisticated initialization of the vertices could be a viable route to explore.

CHAPTER 3

Methods

Having given an overview over the quantities and equations involved, we would like to direct the reader's attention to our proposed method for obtaining a solution.

3.1 Fixed-point structure

We introduce the following functions which are essentially given by eq. (2.21), eq. (2.14) and eq. (2.22):

$$\textsc{parquet}(\Phi) \coloneqq \Lambda + \Phi^{\mathrm{ph}} + \Phi^{\overline{\mathrm{ph}}} + \Phi^{\mathrm{pp}} \tag{3.1a}$$

$$\widetilde{\mathrm{BSE}}(F, \Gamma, G) \coloneqq FGG\Gamma \tag{3.1b}$$

$$\widetilde{\mathrm{SDE}}(F, G) \coloneqq UG(0^-) - UGGFG. \tag{3.1c}$$

The first two are decorated with a tilde because we need to introduce as helper the Dyson equation

$$\textsc{Dyson}(\Sigma) \coloneqq \left(G_0^{-1} - \Sigma\right)^{-1} \tag{3.2}$$

before we are able to write them in a more convenient form

$$\mathrm{BSE}(F, \Phi, \Sigma) \coloneqq \widetilde{\mathrm{BSE}}(F, F - \Phi, \textsc{Dyson}(\Sigma)) \tag{3.3a}$$

$$\mathrm{SDE}(F, \Sigma) \coloneqq \widetilde{\mathrm{SDE}}(F, \textsc{Dyson}(\Sigma)). \tag{3.3b}$$

Then the diagrammatic equations may be written as

$$F = \textsc{parquet}(\Phi) \tag{3.4a}$$

$$\Phi = \mathrm{BSE}(F, \Phi, \Sigma) \tag{3.4b}$$

$$\Sigma = \mathrm{SDE}(F, \Sigma). \tag{3.4c}$$

3.1.1 Traditional method

In previous works (e.g. [23]), solving them has been done mostly via the so-called parquet method (algorithm 1).

Algorithm 1 Parquet method

$\Lambda \leftarrow$ some approximation
$G_0 \leftarrow$ exact expression
$F^{(0)},\ \Phi^{(0)},\ \Sigma^{(0)} \leftarrow \Lambda,\ 0,\ 0$
$\Gamma^{(0)} \leftarrow F^{(0)} - \Phi^{(0)}$
$G^{(0)} \leftarrow G_0$
$i \leftarrow 0$
while $F^{(i)}$, $\Sigma^{(i)}$ not converged **do**
  $i \leftarrow i + 1$
  $\Phi^{(i)} \leftarrow \widetilde{\mathrm{BSE}}\left(F^{(i-1)}, \Gamma^{(i-1)}, G^{(i-1)}\right)$
  $F^{(i)} \leftarrow \textsc{parquet}(\Phi^{(i-1)})$ using Λ
  $\Gamma^{(i)} \leftarrow F^{(i)} - \Phi^{(i)}$
  $\Sigma^{(i)} \leftarrow \widetilde{\mathrm{SDE}}\left(F^{(i)}, G^{(i-1)}\right)$
  $G^{(i)} \leftarrow \textsc{Dyson}(\Sigma^{(i)})$ using G_0

3.1.2 Proposed method

This works well enough, but what we propose instead is trying to exploit the structure more: Simultaneously solving eqs. (3.4a) to (3.4c) constitutes a fixed point problem as sketched out in algorithm 2[1]. Hence, we should be able to benefit from existing research into algorithms for solving such problems.

For choosing an initial point for the fixed-point search, we stick with the traditional approach of using the leading order terms in the quantities' respective U-perturbation expansions. It remains to be investigated how sensitive the results are to this choice of initialization.

Regarding initialization, the stated choices are straightforward to implement and empirically lead to correct solutions in the tested cases, but as discussed above might warrant deeper investigation.

For solving the so-constructed fixed-point problem, we choose the Anderson acceleration algorithm [29] given in algorithm 3 with $m = 5$, which accelerates convergence [30] compared to naive fixed-point iteration. The specific implementation was utilized is the one from `SIAMFANLEquations.jl` [31] with minor tweaks for more verbose conver-

[1] There also may exist an interesting parallel to the multiloop FRG approach of [28].

Algorithm 2 Proposed method

$\Lambda \leftarrow$ some approximation
$G_0 \leftarrow$ exact expression
$F_{\text{init}},\ \Phi_{\text{init}},\ \Sigma_{\text{init}} \leftarrow \Lambda,\ 0,\ 0$
$x_{\text{init}} \leftarrow$ flatten and concatenate $F_{\text{init}},\ \Phi_{\text{init}},\ \Sigma_{\text{init}}$
function DiagrammaticEquations(x)
    $F,\ \Phi,\ \Sigma \leftarrow$ unpack x
    $F' \leftarrow$ parquet(Φ) using Λ
    $\Phi' \leftarrow \text{BSE}(F, \Phi, \Sigma)$ using G_0
    $\Sigma' \leftarrow \text{SDE}(F, \Sigma)$ using G_0
    $x' \leftarrow$ flatten and concatenate $F',\ \Phi',\ \Sigma'$
    return x'
find fixed point of DiagrammaticEquations starting from x_{init}

Algorithm 3 Anderson acceleration

Given a function $f : \mathbb{R}^n \to \mathbb{R}^n$ we want to find a fixed point $x^* = f(x^*)$ of, an initial guess $x_0 \in \mathbb{R}^n$ and an integer parameter $m \geq 1$, define the residual $g(x) = f(x) - x$.
$x_1 \leftarrow f(x_0)$
for $i = 1, 2, 3, \ldots$ **do**
    $m_i \leftarrow \min(m, i)$
    $G \leftarrow \begin{pmatrix} g(x_{i-m_i}) & \ldots & g(x_i) \end{pmatrix}$
    $\alpha \leftarrow \arg\min_{\alpha \in A} \|G\alpha\|_2$ where $A = \{(\alpha_0, \ldots, \alpha_{m_i}) \in \mathbb{R}^{m_i+1} \mid \sum_{k=0}^{m_i} \alpha_k = 1\}$
    $x_{i+1} \leftarrow \sum_{k=0}^{m_i} \alpha_k f(x_{i-m_i+k})$

gence data logging during iteration as well as an adapted convergence criterion, described below.

Figure 3.1 shows a comparison of the convergence histories of the two methods in a typical parameter configuration and clearly demonstrates the superiority of the proposed method in terms of convergence speed. Both methods compute the same functions for each iteration, but the proposed method incurs some additional overhead for whatever the fixed-point solver does internally. Depending on the solver used, compared to the cost of computing the parquet equations, in almost all cases the overhead will be negligible.

Termination condition

The iteration is terminated in case the current residuum $\|x_i - f(x_i)\|$ exceeds the one from $n_{\text{stagnation}}$ iterations ago. This condition reliably detects convergence, with increased $n_{\text{stagnation}}$ trading precision (stopping as early as possible) for robustness (low likelihood

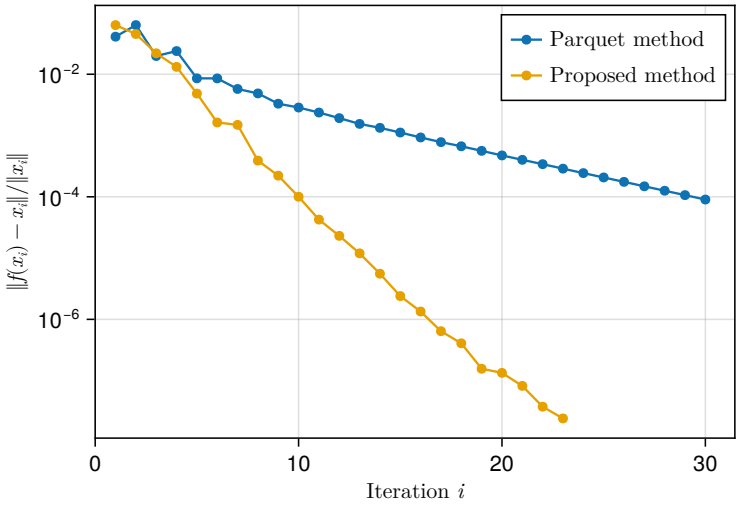

Figure 3.1: Comparison of convergence histories between algorithm 1 and algorithm 2 for the Hubbard model on 4×4 points with $U = 1$, $\beta = 1.2$ and basis cutoff $\varepsilon = 10^{-3}$. Here, Anderson acceleration is used and f denotes DiagrammaticEquations as defined in algorithm 2.

of false positive stopping). An exemplary convergence history is shown in fig. 4.1 below (with termination condition deactivated).

3.2 Representation of quantities

One of the first things to think about when implementing anything related to the parquet method is how to suitably represent the quantities involved. As we have seen, one-particle objects carry two indices, e.g. $\Sigma^{\nu}_{\boldsymbol{k}}$, and two-particle objects carry six indices, e.g. $\Gamma^{\nu\nu'\omega}_{\boldsymbol{k}\boldsymbol{k}'\boldsymbol{q}}$. So especially the two-particle objects require lots of memory (and, by extension, computation time), which we will try to keep to a minimum.

3.2.1 Momentum indices

Concerning the momenta $\boldsymbol{k}$, they are sampled from a regular grid covering the Brillouin zone (BZ). In d dimensions and for the hypercube lattice – to which we will restrict ourselves here – the Brillouin zone is given by $\mathrm{BZ}_d = [0, 2\pi)^d$[2,3]. We take $\boldsymbol{k} \in K_n \coloneqq (0 : 2\pi/n : 2\pi - 2\pi/n)^d$, giving a grid of n^d points[4] [32].

[2]Here we have set the lattice constant a to unity, effectively fixing a system of units.

[3]To be precise, the Brillouin zone is actually $\mathrm{BZ}_d = T^d \equiv S^1 \times \ldots \times S^1$, the d-dimensional torus, manifestly incorporating all physical functions' periodicity.

[4]The syntax $(a : b : c)$ denotes a range of evenly spaced (with step b) values between a and c, or more precisely $\{x \mid a \leq x \leq c \text{ and } x = a + nb \text{ with } n \in \mathbb{Z}\}$.

The hypercube lattice satisfies a particular group of symmetries, the d-dimensional hyperoctahedral group B_d. Considered as a matrix group[5], this is the group of all orthogonal $d \times d$-matrices with integer entries, i.e.

$$B_d = \left\{ o \in \mathbb{Z}^{d \times d} \mid o^\intercal o = I \right\}. \tag{3.5}$$

Any "physically relevant" function of one or more momentum argument(s) $f(\boldsymbol{k}, \boldsymbol{k}', \boldsymbol{k}'', \ldots)$ must therefore be invariant under an appropriate group action, i.e. obey

$$f(\boldsymbol{k}, \boldsymbol{k}', \boldsymbol{k}'', \ldots) = f(o\boldsymbol{k}, o\boldsymbol{k}', o\boldsymbol{k}'', \ldots) \quad \forall o \in B_d, \tag{3.6}$$

yielding an opportunity for storing it in compressed form.

For example a unary function of momentum $f(\boldsymbol{k})$ is fully defined by specifying its values on B_d's quotient BZ_d / B_d, the set of all orbits[6] under the group action. This set is usually referred to as the irreducible Brillouin zone (IBZ) [33] and is isomorphic to a set consisting of a single, arbitrarily chosen representative for each orbit, for example

$$\mathrm{IBZ}_d \sim \{ \boldsymbol{k} \in \mathrm{BZ}_d \mid k_d \leq k_{d-1} \leq \ldots \leq k_1 \leq \pi \}. \tag{3.7}$$

In the following we will use the actual IBZ (the set of orbits) and the so-defined set of representatives interchangeably, keeping in mind that they are not the same thing, strictly speaking. The IBZ's volume is $|\mathrm{IBZ}_d| = |\mathrm{BZ}_d| / |B_d| = |\mathrm{BZ}_d| / (2^d d!)$, so in 2 dimensions we obtain a compression by a factor of up to[7] 8. We call the discretized IBZ $Q_n \coloneqq K_n \cap \mathrm{IBZ}_d$.

The other case appearing in our application are functions of three momentum arguments, often denoted $f(\boldsymbol{k}, \boldsymbol{k}', \boldsymbol{q}')$. Here, our implementation actually does not fully exhaust the potential for compression[8] but instead reduces only the third argument to the IBZ. The resulting momentum grid is shown in fig. 3.2.

Momentum integrals

Approximating momentum integrals is fairly straightforward and has surprisingly agreeable properties. To approximate $I(f) \coloneqq \int_{\mathrm{BZ}_d} f(\boldsymbol{k}) \, \mathrm{d}\boldsymbol{k}$ we use the operator

$$I_n(f) \coloneqq \frac{(2\pi)^d}{n^d} \sum_{\boldsymbol{k}_i \in K_n} f(\boldsymbol{k}_i). \tag{3.8}$$

[5] To illustrate, we have $B_2 = \left\{ \left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix}\right), \left(\begin{smallmatrix} -1 & 0 \\ 0 & 1 \end{smallmatrix}\right), \left(\begin{smallmatrix} 1 & 0 \\ 0 & -1 \end{smallmatrix}\right), \left(\begin{smallmatrix} -1 & 0 \\ 0 & -1 \end{smallmatrix}\right), \left(\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right), \left(\begin{smallmatrix} 0 & -1 \\ 1 & 0 \end{smallmatrix}\right), \left(\begin{smallmatrix} 0 & 1 \\ -1 & 0 \end{smallmatrix}\right), \left(\begin{smallmatrix} 0 & -1 \\ -1 & 0 \end{smallmatrix}\right) \right\}$.

[6] A note on terminology: Given a group G that is acting on a set X, the *orbit* of an element $x \in X$ is the set of all elements of X reachable from x. It is denoted by $Gx = \{ gx \mid g \in G \} \subseteq X$. This notion then allows us to "divide out" the group: The group action's *quotient* X/G is the set of all orbits.

[7] Only "up to" because for the discretized domain the reduction is lower, depending on n. For instance, in the example shown in fig. 3.2, $n = 6$, the compression factor is $6^2/10 = 3.6$.

[8] This is however straightforwardly possible by storing the function only on ${\mathrm{BZ}_d}^3 / B_d$ and should be explored.

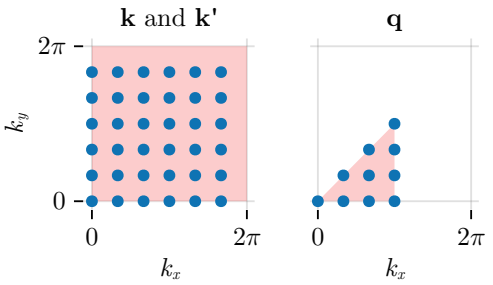

Figure 3.2: The distribution of momentum sampling points on the Brillouin zone in the case of a 6×6 grid. The BZ and the IBZ are highlighted in red.

In practice, we will want to compute integrals on the IBZ alone, for which weight factors according to each point's multiplicity will be required, see below for more. Computing useful bounds on the error $|I - I_n|$ turns out to be quite involved, principally depending on f's smoothness. For example, if f is analytic, convergence is geometric [34], many classes of non-periodic rougher functions are covered in [35].

IBZ-reduced momentum integrals Concerning integrals over IBZ-reduced momentum indices, e.g. the bosonic index in 2-particle quantities, we need to know the multiplicity $m_{\boldsymbol{q}}$ of each point $\boldsymbol{q}$, which is the number of points in the entire BZ it is equivalent to[9], i.e.

$$m_{\boldsymbol{q}} \coloneqq |\{\boldsymbol{k} \in \mathrm{BZ}_d \mid \exists o \in B_d,\ o\boldsymbol{k} = \boldsymbol{q}\}| = |B_d \boldsymbol{q}|. \tag{3.9}$$

We then modify eq. (3.8) to weigh each term by $m_{\boldsymbol{q}_i}$ and get

$$I_n^{\mathrm{IBZ}_d}(f) \coloneqq \frac{(2\pi)^d}{n^d} \sum_{\boldsymbol{q}_i \in Q_n} m_{\boldsymbol{q}_i} f(\boldsymbol{q}_i). \tag{3.10}$$

Multiple momentum arguments integrals The case of multiple momentum arguments requires special consideration. Let f be a function of two momentum arguments and compress it as previously discussed, i.e. store the second argument only on IBZ points to get $f_{\mathrm{red}}(\boldsymbol{k}, \boldsymbol{q})$ where $\boldsymbol{k} \in \mathrm{BZ}_d$ and $\boldsymbol{q} \in \mathrm{IBZ}_d$. Then, to approximate the integral over the second argument $I(f)(\boldsymbol{k}) \coloneqq \int_{\mathrm{BZ}_d} f(\boldsymbol{k}, \boldsymbol{k}')\, \mathrm{d}\boldsymbol{k}'$ we use

$$I_n(f)(\boldsymbol{k}) = \frac{(2\pi)^d}{n^d} \sum_{\boldsymbol{k}'_i \in K_n} f(\boldsymbol{k}, \boldsymbol{k}'_i) = \frac{(2\pi)^d}{n^d} \sum_{\boldsymbol{k}'_i \in K_n} f_{\mathrm{red}}(o_i \boldsymbol{k}, o_i \boldsymbol{k}'_i), \tag{3.11}$$

where $o_i \in B_d$ is any transformation such that $o_i \boldsymbol{k}'_i \in \mathrm{IBZ}_d$ (here IBZ_d is to be interpreted as the set arbitrarily chosen as representatives). Depending on $\boldsymbol{k}'_i$, there may be multiple

[9] This is also called the length of the orbit of $\boldsymbol{q}$ under B_d.

choices of o_i (all of which are valid), reflecting a not-yet-exploited potential for further compressing the vertices. This potential could be realized by properly reducing $\mathrm{BZ}_d{}^3$ by B_d.

3.2.2 Frequency indices

As for the Matsubara frequencies, here the situation is more interesting. A first, natural optimization is only storing vertices on frequencies (ν, ν', ω) with $\omega \geq 0$, enabled by the property

$$G^{\nu\nu'\omega}_{\boldsymbol{k}\boldsymbol{k}'\boldsymbol{q}} = \left(G^{(-\nu)(-\nu')(-\omega)}_{\boldsymbol{k}\boldsymbol{k}'\boldsymbol{q}}\right)^*, \tag{3.12}$$

which holds as long as H is real. Traditionally, a dense box of frequencies is used, incurring undesirable memory and runtime scaling. Recent developments allow us to significantly improve on this. Using the intermediate representation developed by Shinaoka et al. in [24] and implemented in [36] enables a much reduced, sparse sampling set. For example, in setting up for the $U = 1t$, $\beta t = 5$ Hubbard model problem at $\varepsilon = 10^{-3}$ precision, we construct a sparse sampling set of only 2545 frequencies, visualized in fig. 3.3, whereas for example a $64 \times 64 \times 64$ traditional frequency box[10] already means 262144 frequencies, over 2 orders of magnitude more!

3.2.3 Intermediate representation

Originally published in [24], there is a compression method based on an intermediate representation for the objects we are interested in and that are fundamental to the present work. We will here present a short version of the more comprehensive treatment provided in [37]. To match the original source [37], in this section we will use a slightly modified notation for Matsubara frequencies, writing $\mathrm{i}\nu$ instead of ν to emphasize the imaginary nature.

Two-point quantities

The fermionic ($\alpha = \mathrm{F}$) respectively bosonic ($\alpha = \mathrm{B}$) Green's function on Matsubara frequencies $G^\alpha(\mathrm{i}\omega)$ is related to the spectral function in real frequency $\rho^\alpha(\omega)$ by

$$G^\alpha(\mathrm{i}\omega) = \int_{-\omega_{\max}}^{+\omega_{\max}} \mathrm{d}\omega'\, K^\alpha(\mathrm{i}\omega, \omega')\rho^\alpha(\omega') \tag{3.13}$$

with kernels $K^{\mathrm{F}}(\mathrm{i}\omega, \omega) = \frac{1}{\mathrm{i}\omega - \omega'}$ and $K^{\mathrm{B}}(\mathrm{i}\omega, \omega) = \frac{\omega'}{\mathrm{i}\omega - \omega'}$. Here, $\omega_{\max}$ denotes the bandwidth, i.e. we assume that ρ^α's support is contained in $[-\omega_{\max}, \omega_{\max}]$ Analogous to a matrix' singular value decomposition, there is a singular value expansion for this kernel

$$K^\alpha(\mathrm{i}\omega, \omega) = \sum_{l=0}^{\infty} U_l^\alpha(\mathrm{i}\omega) S_l^\alpha V_l^\alpha(\omega) \tag{3.14}$$

[10] As used in the reference computation presented below.

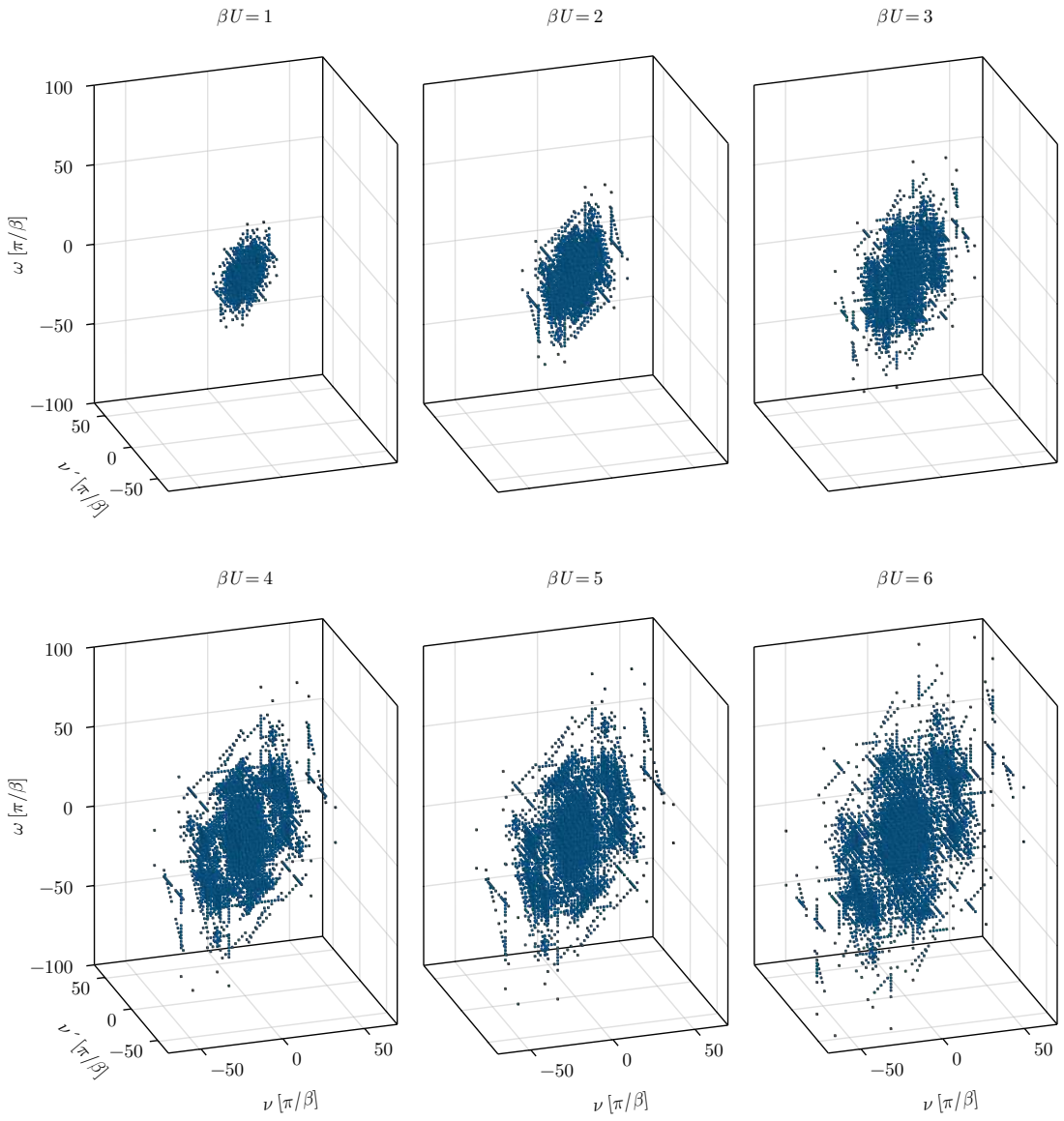

Figure 3.3: The (ν, ν', ω) frequency sampling points on which all two-particle quantities are represented. These points are generated for bandwidth $4U$ and tolerance $\varepsilon = 10^{-3}$. At lower temperatures more frequencies are needed.

that, inserted into eq. (3.13) yields a representation

$$G^\alpha(\mathrm{i}\omega) = \sum_{l=0}^{\infty} U_l^\alpha(\mathrm{i}\omega) G_l^\alpha \tag{3.15}$$

of the Green's function with coefficients

$$G_l^\alpha = S_l^\alpha \int_{-\omega_{\max}}^{+\omega_{\max}} \mathrm{d}\omega'\, V_l^\alpha(\omega')\rho^\alpha(\omega'). \tag{3.16}$$

We may now choose a cutoff ε for the singular values to get a compressed, approximate representation of G using L coefficients, such that $S_L/S_0 < \varepsilon$. Although this compression is lossy, it is optimally compact [36].

To compute the coefficients from $G(\mathrm{i}\omega)$, we sample at a number of sampling frequencies

$$\mathcal{W}^\alpha = \{\mathrm{i}\omega_1^\alpha, \mathrm{i}\omega_2^\alpha, \ldots, \mathrm{i}\omega_L^\alpha\}, \tag{3.17}$$

chosen close to the sign changes of $U_L^\alpha(\mathrm{i}\omega)$, and perform a least-squares fit

$$\{G_l^\alpha\} = \underset{\{G_l^\alpha\}}{\arg\min} \sum_{\mathrm{i}\omega \in \mathcal{W}^\alpha} \left| G^\alpha(\mathrm{i}\omega) - \sum_{l=0}^{L-1} U_l^\alpha(\mathrm{i}\omega) G_l^\alpha \right|^2. \tag{3.18}$$

The set of sampling frequencies $\mathcal{W}^\alpha$ is symmetric about zero, i.e. $\mathcal{W}^\alpha = -\mathcal{W}^\alpha$.

Four-point quantities

For the representation of four-point quantities we define

$$U_{ll'm}^3((\mathrm{i}\nu, \mathrm{i}\nu', \mathrm{i}\omega)) \equiv U_l^\mathrm{F}(\mathrm{i}\nu) U_{l'}^\mathrm{F}(\mathrm{i}\nu') U_l^\mathrm{B}(\mathrm{i}\omega), \tag{3.19}$$

whose naive usage in generalizing eq. (3.15) turns out to be invalid [38, 39], instead requiring an overcomplete representation

$$G(\mathrm{i}\nu_1, \ldots, \mathrm{i}\nu_4) \approx \sum_{r=1}^{12} \sum_{l,l',m=0}^{L-1} U_{ll'm}^3(T_r(\mathrm{i}\nu_1, \ldots, \mathrm{i}\nu_4)) G_{r,ll'm}. \tag{3.20}$$

The T_r are a family of 12 frequency translation functions given by

$$T_r = C^{\to 3} \circ P_r, \tag{3.21}$$

i.e. composing conversion to four-frequency notation

$$C^{\to 3}(\nu_1, \ldots, \nu_4) = (\nu_1, -\nu_4, \nu_1 + \nu_2) \tag{3.22}$$

with a permutation as defined in table 3.1.

This yields a sampling set of

$$\mathcal{W} = \bigcup_{r=1}^{12} T_r^{-1}\Big(\mathcal{W}^\mathrm{F} \times \mathcal{W}^\mathrm{F} \times \mathcal{W}^\mathrm{B}\Big). \tag{3.23}$$

Associated with this construction is yet again a requested tolerance ε that is used for truncating the constituent fermionic and bosonic singular value expansions, i.e.

$$S_L^\mathrm{F}/S_0^\mathrm{F} < \varepsilon \quad \text{and} \quad S_L^\mathrm{B}/S_0^\mathrm{B} < \varepsilon. \tag{3.24}$$

r	P_r	
	Cycle notation	Image of string 1234
1	()	1234
2	(3, 4)	1243
3	(2, 3)	1324
4	(2, 3, 4)	1342
5	(2, 4, 3)	1423
6	(2, 4)	1432
7	(1, 2)	2134
8	(1, 2)(3, 4)	2143
9	(1, 2, 3)	2314
10	(1, 2, 4, 3)	2413
11	(1, 3, 2)	3124
12	(1, 3)	3214

Table 3.1: The permutations used in constructing the two-particle overcomplete basis [39].

Sparse frequency set closed under frequency permutations The so-constructed set $\mathcal{W}$ has the useful property of being closed under any permutation of the four frequencies. That is, for any permutation $P \in S_4$ (with S_4 the symmetric group of degree 4, i.e. the group of all permutations on a set of size four) and any element $\boldsymbol{\nu} \in \mathcal{W}$, $P\boldsymbol{\nu} \in \mathcal{W}$, as we will now show.

For brevity, we introduce the notation

$$\mathcal{W}_4 = C^{\to 4}\Big(\mathcal{W}^{\mathrm{F}} \times \mathcal{W}^{\mathrm{F}} \times \mathcal{W}^{\mathrm{B}}\Big), \tag{3.25}$$

where $C^{\to 4} \circ C^{\to 3} = \text{identity}$, i.e. $C^{\to 4}(\nu, \nu', \omega) = (\nu, -\nu+\omega, \nu'-\omega, -\nu')$, so that eq. (3.23) may be written

$$\mathcal{W} = \bigcup_{r=1}^{12} P_r^{-1}(\mathcal{W}_4). \tag{3.26}$$

The set $\mathcal{W}_4$ is invariant under the permutations $(1234 \mapsto 1234) \equiv ()$ (trivially) and $(1234 \mapsto 4321) \equiv (1,4)(2,3)$, as can be seen by exchanging $\nu \leftrightarrow -\nu'$ and $\omega \leftrightarrow -\omega$ in eq. (3.22), and $\mathcal{W} = -\mathcal{W}$ (see above).

Now, consider any $\boldsymbol{\nu} \in \mathcal{W}$; there is at least one r with an associated $\boldsymbol{\nu}_0 \in \mathcal{W}_4$ such that $\boldsymbol{\nu} = P_r^{-1}\boldsymbol{\nu}_0$. If we now pick any $P \in S_4$, we can write

$$P\boldsymbol{\nu} = P\Big(P_r^{-1}\boldsymbol{\nu}_0\Big) = P'\boldsymbol{\nu}_0 \tag{3.27}$$

with some other permutation $P' = P \circ P_r^{-1} \in S_4$. Thus, the following holds:

$$
\begin{aligned}
P'\boldsymbol{\nu}_0 \in \mathcal{W} &\iff \exists r' \quad \text{such that} \quad P'\boldsymbol{\nu}_0 \in P_{r'}^{-1}(\mathcal{W}_4) && \text{(3.28a)}\\
&\iff \exists r' \quad \text{such that} \quad \boldsymbol{\nu}_0 \in P'^{-1}\left(P_{r'}^{-1}(\mathcal{W}_4)\right) && \text{(3.28b)}\\
&\impliedby \exists r' \quad \text{such that} \quad P'^{-1}\left(P_{r'}^{-1}(\mathcal{W}_4)\right) = \mathcal{W}_4 && \text{(3.28c)}\\
&\iff \exists r' \quad \text{such that} \quad P'^{-1} \circ P_{r'}^{-1} = () \quad \text{or} \quad (1,4)(2,3) && \text{(3.28d)}\\
&\iff \exists r' \quad \text{such that} \quad P_{r'} = P' \quad \text{or} \quad (1,4)(2,3) \circ P' && \text{(3.28e)}
\end{aligned}
$$

To summarize, as long as, for any given permutation $P \in S_4$, our set $\{P_r\} \subseteq S_4$ contains either P or $(1,4)(2,3) \circ P$, the sampling frequencies $\mathcal{W}$ are closed with respect to all permutations. Our chosen $\{P_r\}$ satisfies this property; it is in fact one of the $2^{12} = 4096$ minimal subsets of S_4 to do so.

3.3 Parquet equation

The implementation of the parquet equation eq. (2.21) is fairly straightforward. Its most prominent feature are the frequency/momentum shifts it necessitates. These shifts present a considerable problem in the usual treatment of the parquet equation. For example, [23] deals with this problem of shifted frequencies exceeding the frequency box as follows:

> Due to crossing symmetry applied to $\overline{\text{ph}}$-vertices in order to convert them to ph-notation, the second fermionic frequency argument is shifted as $\nu' \to (\nu+\omega)$ and values can exceed the frequency box. `multi-orbital-parquet` overcomes this problem by using Kernel functions which employ the scan-edge method [25, p. 7]: if $\nu + \omega$ exceeds the box, we approximate $\Phi^{\nu(\nu+\omega)(\nu'-\nu)} \approx \Phi^{\nu\tilde{\nu}(\nu'-\nu)}$ by taking the nearest value at the edge, i.e. $\tilde{\nu} = \nu_{-N_F/2}$ or $\tilde{\nu} = \nu_{N_F/2-1}$, as $\Phi^{\nu\nu'\omega}$ is asymptotically constant along ν' for fixed ν and ω. Note that for vertices with bosonic frequencies outside the box no approximation with Kernel functions is provided, they are set to 0.

An unexpected, but all the more welcome, consequence of representing the vertices including Φ on the sparse frequency set introduced in section 3.2.3 is that the frequency shifts in the parquet equation by construction always result in another point in the sparse set. To see this, note that the frequency shifts in three-frequency convention are the result of frequency permutations in (full) four-frequency notation eq. (2.20) and thus the above result concerning the invariance of our frequency sampling set under permutation applies. We hence do not incur any approximation error and our parquet equation's implementation is exact up to floating point errors.

Of course, we need to consider how to correctly implement the momentum shifts in the parquet equation. For every occurrence of $\Phi_{\tilde{\boldsymbol{k}}\tilde{\boldsymbol{k}}'\tilde{\boldsymbol{q}}}$ on the right-hand side $\tilde{\boldsymbol{k}}$, $\tilde{\boldsymbol{k}}'$ and $\tilde{\boldsymbol{q}}$

are the sum or difference of momenta and as such might be outside the BZ. Therefore, all momenta are mapped back to the BZ to yield $\boldsymbol{k}$, $\boldsymbol{k}'$ and $\boldsymbol{q}$. In general the bosonic momentum $\boldsymbol{q}$ which we have restricted to the IBZ might still be outside the IBZ. By construction, there is a unique $o \in B_d$ such that $o\boldsymbol{q} \in \mathrm{IBZ}$, and thus

$$\Phi_{\tilde{\boldsymbol{k}}\tilde{\boldsymbol{k}}'\tilde{\boldsymbol{q}}} = \Phi_{\boldsymbol{k}\boldsymbol{k}'\boldsymbol{q}} = \Phi_{(o\boldsymbol{k})(o\boldsymbol{k}')(o\boldsymbol{q})}. \tag{3.29}$$

The final thing we need to be careful about is negative bosonic frequencies: For the case $\omega < 0$ in $\Phi^{\nu\nu'\omega}$, eq. (3.12) comes to bear, and we use

$$\Phi^{\nu\nu'\omega} = \left(\Phi^{(-\nu)(-\nu')(-\omega)}\right)^*. \tag{3.30}$$

3.4 Bethe-Salpeter equation

The Bethe-Salpeter equation eq. (2.14) revolves around a frequency sum, posing significant challenges.

3.4.1 Separating off the backgrounds

In practice, the sum needs to be truncated

$$\sum_{\nu_1=-\infty}^{\infty} \to \sum_{\nu_1=-\nu_{\max}}^{\nu_{\max}}, \tag{3.31}$$

where the addends decay quadratically

$$\Gamma^{\nu\nu_1\omega}_{\boldsymbol{k}\boldsymbol{k}_1\boldsymbol{q}} X_0{}^{\nu_1\omega}_{\boldsymbol{k}_1\boldsymbol{q}} F^{\nu_1\nu'\omega}_{\boldsymbol{k}_1\boldsymbol{k}'\boldsymbol{q}} = \mathcal{O}\left(\nu_1^{-2}\right), \tag{3.32}$$

incurring an error decaying linearly in $\nu_{\max}$, requiring the evaluation of large frequency sets and in turn slowing down further what is already the bottleneck of the entire program.

To improve on this situation, we make a key simplifying assumption which is not strictly valid, but works as a rough approximation, namely that

$$\Gamma^{r\,\nu\nu_1\omega}_{\boldsymbol{k}\boldsymbol{k}_1\boldsymbol{q}} = U^r + \mathcal{O}\left(\nu_1^{-1}\right), \tag{3.33a}$$

$$\text{and} \quad F^{r\,\nu\nu_1\omega}_{\boldsymbol{k}\boldsymbol{k}_1\boldsymbol{q}} = U^r + \mathcal{O}\left(\nu_1^{-1}\right), \tag{3.33b}$$

which allows us to compute the background part of the sum analytically[11]. To see how this fails to hold, refer to fig. 4.12: It clearly shows the vertex featuring diagonal lines which do not decay. Removing the need for this optimization would be a top priority for future work.

[11] By U^r we mean the bare vertex in a particular channel r, whereas U is the parameter in the Hamiltonian. It is given by $U^{\mathrm{d}} = U$, $U^{\mathrm{m}} = -U$, $U^{\mathrm{s}} = 2U$ and $U^{\mathrm{t}} = 0$.

Nevertheless, we rewrite the sum to

$$\int_{\boldsymbol{k}_1} \sum_{\nu_1} \left(\Gamma^{\nu\nu_1\omega}_{\boldsymbol{k}\boldsymbol{k}_1\boldsymbol{q}} F^{\nu_1\nu'\omega}_{\boldsymbol{k}_1\boldsymbol{k}'\boldsymbol{q}} - U^2 \right) X_0{}^{\nu_1\omega}_{\boldsymbol{k}_1\boldsymbol{q}} + \int_{\boldsymbol{k}_1} \sum_{\nu_1} U^2 X_0{}^{\nu_1\omega}_{\boldsymbol{k}_1\boldsymbol{q}} \tag{3.34}$$

and focus on the second part, the Lindhard bubble

$$B^{r\omega}_{\boldsymbol{q}} := \int_{\boldsymbol{k}_1} \sum_{\nu_1} U^r X_0{}^{r\nu_1\omega}_{\boldsymbol{k}_1\boldsymbol{q}} U^r, \tag{3.35}$$

first.

For the sake of completeness we produce here the calculation for the case $r \in \mathrm{ph}$.

$$\begin{aligned}
B^{\mathrm{ph}\omega}_{\boldsymbol{q}} &= \int_{\boldsymbol{k}_1} \sum_{\nu_1} U^{\mathrm{ph}} X_0{}^{\mathrm{ph}\nu_1\omega}_{\boldsymbol{k}_1\boldsymbol{q}} U^{\mathrm{ph}} \\
&= {U^{\mathrm{ph}}}^2 \int_{\boldsymbol{k}_1} \sum_{\nu_1} \left(-G^{\nu_1}_{\boldsymbol{k}_1} G^{\nu_1+\omega}_{\boldsymbol{k}_1+\boldsymbol{q}} \right) \\
&= {U^{\mathrm{ph}}}^2 \int_{\boldsymbol{k}_1} \sum_{\nu_1} \left(-\int_0^\beta \mathrm{d}\tau\, \mathrm{e}^{-\mathrm{i}\nu_1\tau} \hat{G}_{\boldsymbol{k}_1}(\tau) \int_0^\beta \mathrm{d}\tau'\, \mathrm{e}^{-\mathrm{i}(\nu_1+\omega)\tau'} \hat{G}_{\boldsymbol{k}_1+\boldsymbol{q}}(\tau') \right) \\
&= -{U^{\mathrm{ph}}}^2 \int_{\boldsymbol{k}_1} \int_0^\beta \mathrm{d}\tau \int_0^\beta \mathrm{d}\tau'\, \hat{G}_{\boldsymbol{k}_1}(\tau) \mathrm{e}^{-\mathrm{i}\omega\tau'} \hat{G}_{\boldsymbol{k}_1+\boldsymbol{q}}(\tau') \sum_{\nu_1} \mathrm{e}^{-\mathrm{i}\nu_1(\tau+\tau')} \\
&= -{U^{\mathrm{ph}}}^2 \int_{\boldsymbol{k}_1} \int_0^\beta \mathrm{d}\tau \int_0^\beta \mathrm{d}\tau'\, \hat{G}_{\boldsymbol{k}_1}(\tau) \mathrm{e}^{-\mathrm{i}\omega\tau'} \hat{G}_{\boldsymbol{k}_1+\boldsymbol{q}}(\tau') \beta \sum_{n\in\mathbb{Z}} (-1)^n \delta(\tau+\tau'-n\beta) \\
&= -{U^{\mathrm{ph}}}^2 \int_{\boldsymbol{k}_1} \int_0^\beta \mathrm{d}\tau\, \hat{G}_{\boldsymbol{k}_1}(\tau) \mathrm{e}^{-\mathrm{i}\omega(\beta-\tau)} \hat{G}_{\boldsymbol{k}_1+\boldsymbol{q}}(\beta-\tau) \beta (-1) \\
&= \beta {U^{\mathrm{ph}}}^2 \int_{\boldsymbol{k}_1} \int_0^\beta \mathrm{d}\tau\, \mathrm{e}^{\mathrm{i}\omega\tau} \hat{G}_{\boldsymbol{k}_1}(\tau) \hat{G}_{\boldsymbol{k}_1+\boldsymbol{q}}(\beta-\tau)
\end{aligned}$$

So we have computed the Fourier transformed bubble

$$\widehat{B^{\mathrm{ph}}}_{\boldsymbol{q}}(\tau) = \beta {U^{\mathrm{ph}}}^2 \int_{\boldsymbol{k}_1} \hat{G}_{\boldsymbol{k}_1}(\beta-\tau) \hat{G}_{\boldsymbol{k}_1+\boldsymbol{q}}(\tau), \tag{3.36}$$

at which point the intermediate representation makes a surprise appearance; We only need to compute $\widehat{B^{\mathrm{ph}}}_{\boldsymbol{q}}$ at a few τ sampling locations, fit the result to a 1-particle basis and expand at the needed bosonic frequencies ω as detailed in section 3.2.3. This allows us to calculate the bubble without truncating the original Matsubara sum or expensively approximating the Fourier transform. The only error we make comes from truncating the basis, the cutoff for which we can choose to machine precision as these one-particle computations are negligible from a runtime cost viewpoint regardless.

With this done, only the first term of eq. (3.34) remains. Because we have subtracted the backgrounds of F and Γ, the addend is now $\mathcal{O}\left(\nu_1^{-3}\right)$ and the sum will converge quadratically. This new sum we do indeed just truncate to the frequencies from $-\nu_{\max}$ to $\nu_{\max}$ but are able to choose a much smaller cutoff $\nu_{\max}$, which currently must be given as input to the algorithm, thanks to calculating the background's contribution separately.

3.4.2 Summation frequencies

Having somewhat mitigated the convergence issue, we turn our attention to a second problem. Because we know the BSE's input vertices – Γ and F – only at some frequencies, the sparse sampling set, we cannot straightforwardly evaluate the (truncated) Matsubara sum.

To compute $\Phi^{r\nu\nu'\omega}_{\boldsymbol{k}\boldsymbol{k}'\boldsymbol{q}}$ at one point in frequency space, we need to know $\Gamma^{r\nu\nu_1\omega}_{\boldsymbol{k}\boldsymbol{k}_1\boldsymbol{q}}$ and $F^{r\nu_1\nu'\omega}_{\boldsymbol{k}_1\boldsymbol{k}'\boldsymbol{q}}$ for *all* $\nu_1 \in (-\nu_{\max} : 2\pi/\beta : \nu_{\max})$, which we do not, as this exceeds the sparse sampling set. So the only option we have is to fit the entire vertices Γ and F to a 2-particle basis and expand at the much larger sets indicated prior to performing the summation.

This single step – more precisely, the expansion substep – actually turns out to be the bottleneck for the implementation of the entire algorithm. Several rounds of optimizations – including prefactorizing the fit matrix, parallelizing the expansion operator at significant memory cost and mostly eliminating allocations – were not able to change this fact.

There is the possibility of computing a set of weights to enable carrying out the BSE sum on the sparse sampling set directly [37], which we forwent as there remain questions about the error this method incurs that remain to be answered.

3.5 Schwinger-Dyson equation

Our implementation of the Schwinger-Dyson equation actually utilizes many of the already mentioned techniques, so we will not discuss them here in detail. One low-hanging fruit is the optimized computing of the Schwinger-Dyson equation: We only compute the self-energy on positive Matsubara frequencies and then use

$$\Sigma_{\boldsymbol{k}}(\nu) = [\Sigma_{\boldsymbol{k}}(-\nu)]^*. \tag{3.37}$$

Also, analogously to our treatment of the Bethe-Salpeter equation in section 3.4.1 we can substantially improve the SDE's numerical properties by subtracting from F its frequency-constant diagram U, and computing the two resulting terms separately.

CHAPTER 4

Results

We would now like to discuss and analyze the results of the presented method. All results have been generated with Julia `1.10.0-rc1` using following packages:

- `SparseIR.jl` `1.0.15`
- `OvercompleteIR.jl` (unreleased)
- `StaticArrays.jl` `1.7.0`
- `LowRankApprox.jl` `0.5.4`

Additionally, as briefly stated above, we employ the Anderson acceleration implementation from [31] for solving the fixed-point problem. We opted to copy the source code into our project, allowing us to make minor customizations – such as more verbose iteration logging or custom termination criteria – that are not suitable for upstream contribution, rather than depending on the original package. For the purposes of this chapter, the function f is shorthand for DiagrammaticEquations defined in algorithm 2.

Any norm used in the following is the Euclidean one taken over all respective frequency and momentum sampling points.

4.1 Convergence properties

In order to gain some measure of confidence in the discussed method, we analyze its convergence properties.

4.1.1 Single-run convergence

First we investigate the convergence history of solving two models,

- Hubbard atom with $U = 1t$ and $\beta = 5/t$ and
- 2d Hubbard model with $U = 1t$ and $\beta = 5/t$ on 4×4 $\boldsymbol{k}$-points,

both solved with BSE sum cutoff $\nu_{\max} = 20$ and overcomplete basis cutoff $\varepsilon = 10^{-3}$, as defined in eq. (3.24). The fixed-point residuum is displayed in fig. 4.1.

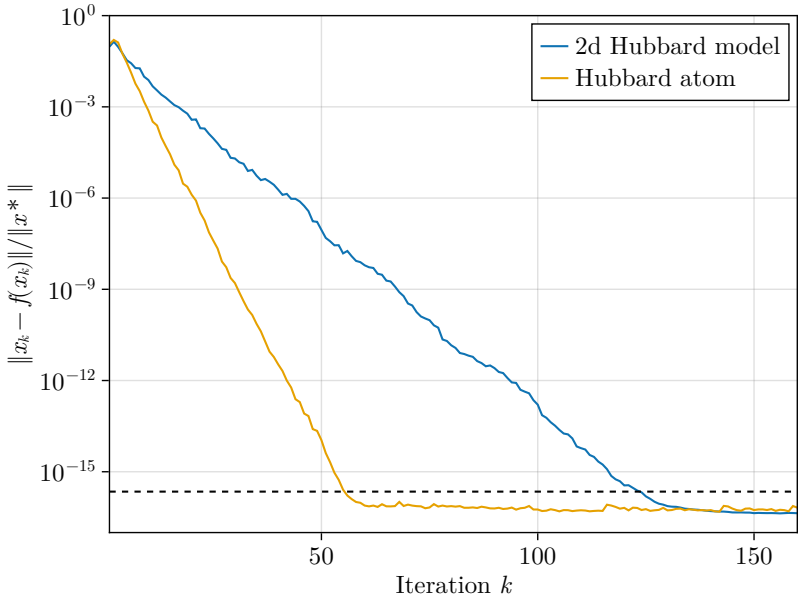

Figure 4.1: The fixed-point solver's typical convergence history, displayed as residua over the course of the entire solution process for two models. x_k is the discussed, kth iteration state vector which comprises the flattened and concatenated vertices F, Φ and Σ. x^* denotes f's (empirical) fixed-point. The dashed line indicates double precision machine epsilon 2^{-52}.

It exhibits quite favorable properties: the convergence rate is exponential, followed by a stable noise floor below machine precision $\varepsilon_{\text{MP}} = 2^{-52}$. This of course does not constitute proof that the method converges at all points in the parameter region, but all tested parameters work well and did not give reason for concern.

4.1.2 Convergence with tolerance and Matsubara sum cutoff

It is now imperative to check for convergence with decreasing tolerance ε (c.f. section 3.2.3) and increasing Matsubara cutoff $\nu_{\max}$ (eq. (3.31)). For this purpose we first investigate the atomic limit, where an analytic solution is available [22] to compare to.

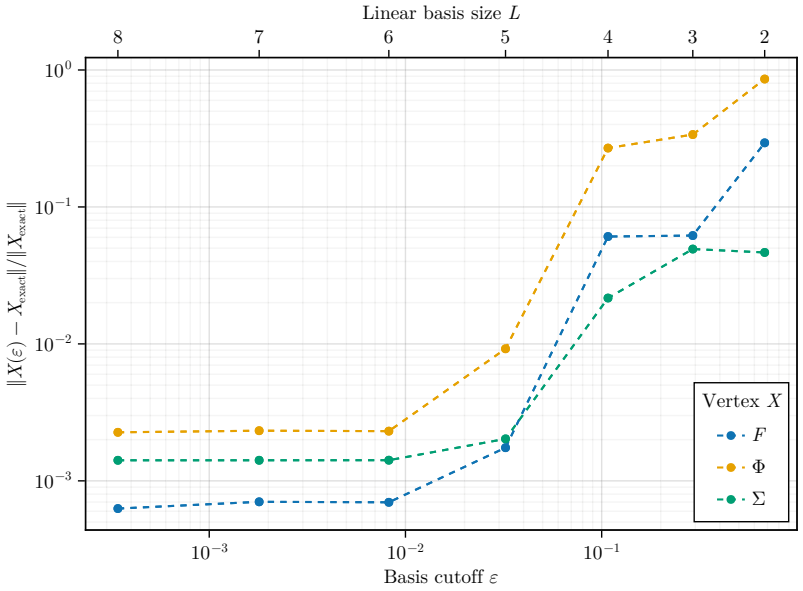

Figure 4.2: Atomic limit: Error in the three vertices $X(\varepsilon, \nu_{\max})$ – with $X \in \{F, \Phi, \Sigma\}$. As benchmark reference we take the analytic solutions X_{exact} from [22]. The vertices are approximate fixed-point solutions to the half-filled Hubbard atom with $U = 1.2$ and $\beta = 2.3$. Here, ε is varied and $\nu_{\max} = 199$ is set for all computations.

For fixed $\nu_{\max}$ (fig. 4.2) we see polynomial convergence somewhat faster than $\mathcal{O}(\varepsilon^{-1})$ before hitting a floor around 10^{-3}, with the exact limiting value depending on the specific vertex we look at.

Fixing ε, we see a somewhat surprising scaling behavior: in fig. 4.3 the self-energy Σ hits an error floor sooner than the four-point vertices F and Φ. This seems to indicate a systematic error somewhere in our approach, and we strongly suspect the U^2-correction eq. (3.33a) as likely culprit.

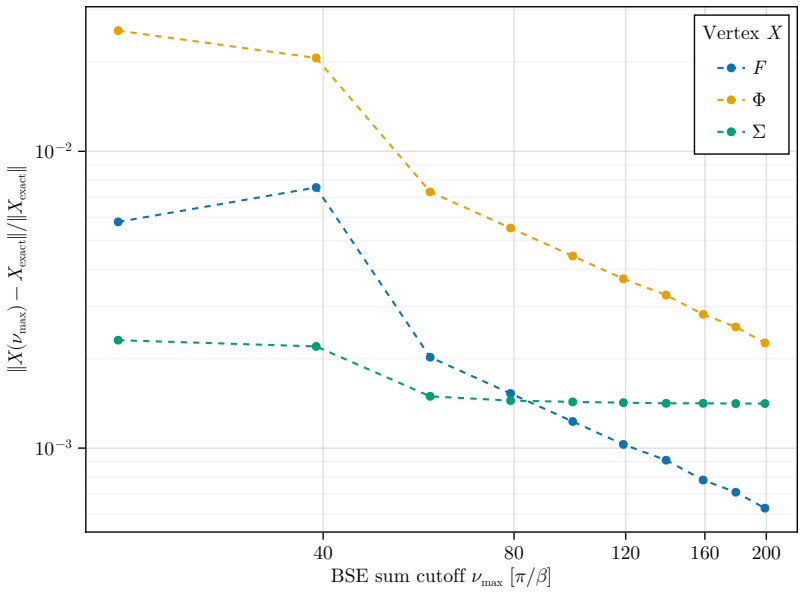

Figure 4.3: Atomic limit: Same as fig. 4.2, but $\nu_{\max}$ is varied and $\varepsilon \approx 3.4 \times 10^{-4}$ is set for all computations.

Regarding the Hubbard model on 4×4 momentum points, the solution resulting from the smallest ε and largest $\nu_{\max}$ will be the most accurate one, so we take it as the reference solution to compare the others to. The results are then shown as a function of ε in fig. 4.4 with fixed $\nu_{\max}$ and as a function of ε in fig. 4.5 with fixed ε.

In this case, there is no apparent error floor which makes sense considering the reference we compare to is bound to be "systematically similar" to the result evaluated. Scaling-wise, in fig. 4.4 we again observe slightly-better-than-$\mathcal{O}(\varepsilon)$ polynomial behavior[1], while fig. 4.5 exhibits super-polynomial scaling regarding the frequency cutoff $\nu_{\max}$.

[1] For example, decreasing the basis cutoff by a factor ~ 225 from the rightmost to the leftmost data point, the error in Φ – which scaled the worst out of the three vertices – decreased by a factor of ~ 545.

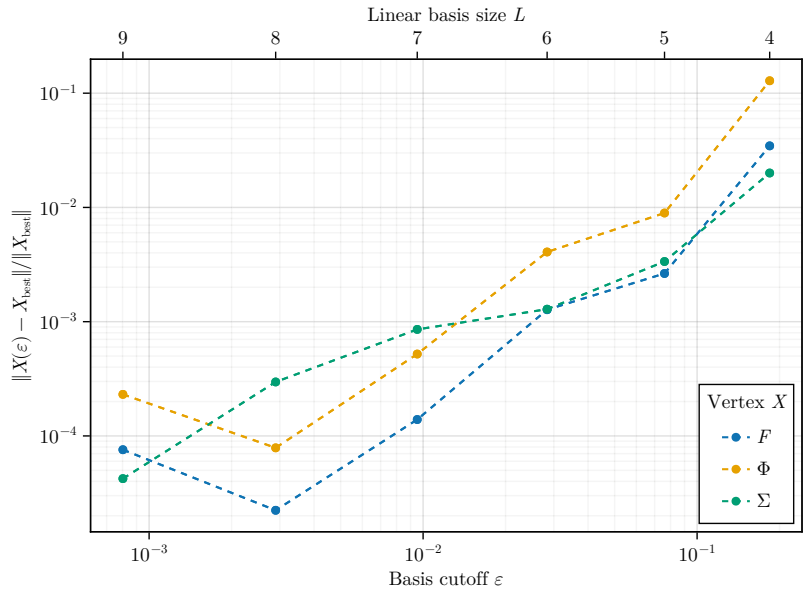

Figure 4.4: Hubbard model: Error in the three vertices $X(\varepsilon, \nu_{\max})$ – with $X \in \{F, \Phi, \Sigma\}$. As benchmark reference we take $X_{\text{best}} \equiv X(3.4 \times 10^{-4}, 199\pi/\beta)$. The vertices are approximate fixed-point solutions to the half-filled Hubbard model with $U = 1t$ and $\beta = 5/t$ on 4×4 $\boldsymbol{k}$-points. Here, ε is varied and $\nu_{\max} = 199$ is set for all computations.

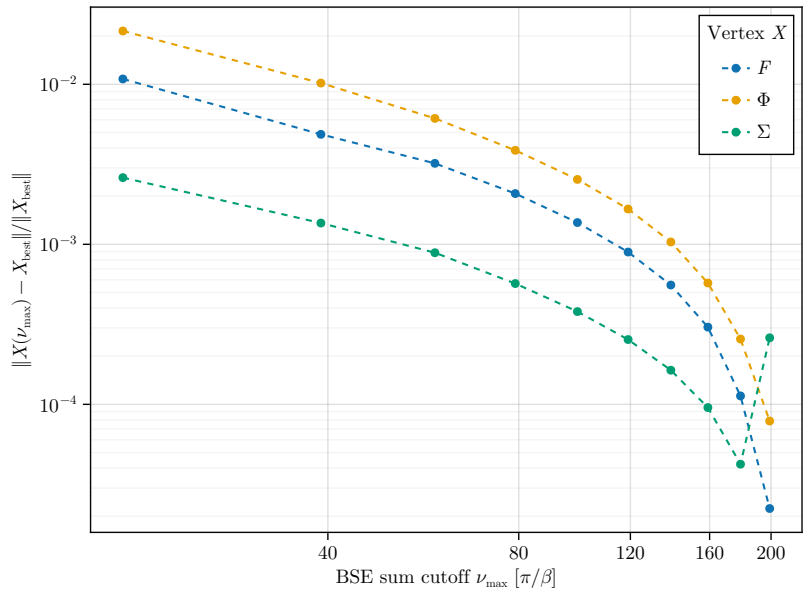

Figure 4.5: Same as fig. 4.4, but now $\nu_{\max}$ is varied and $\varepsilon \approx 3.4 \times 10^{-4}$ is fixed for all computations.

4.2 Benchmarks

To evaluate our code's performance, we conduct a series of benchmarks of representative workloads. First we investigate the two precision parameters' ε and $\nu_{\max}$ effect on runtime, visualized in fig. 4.6.

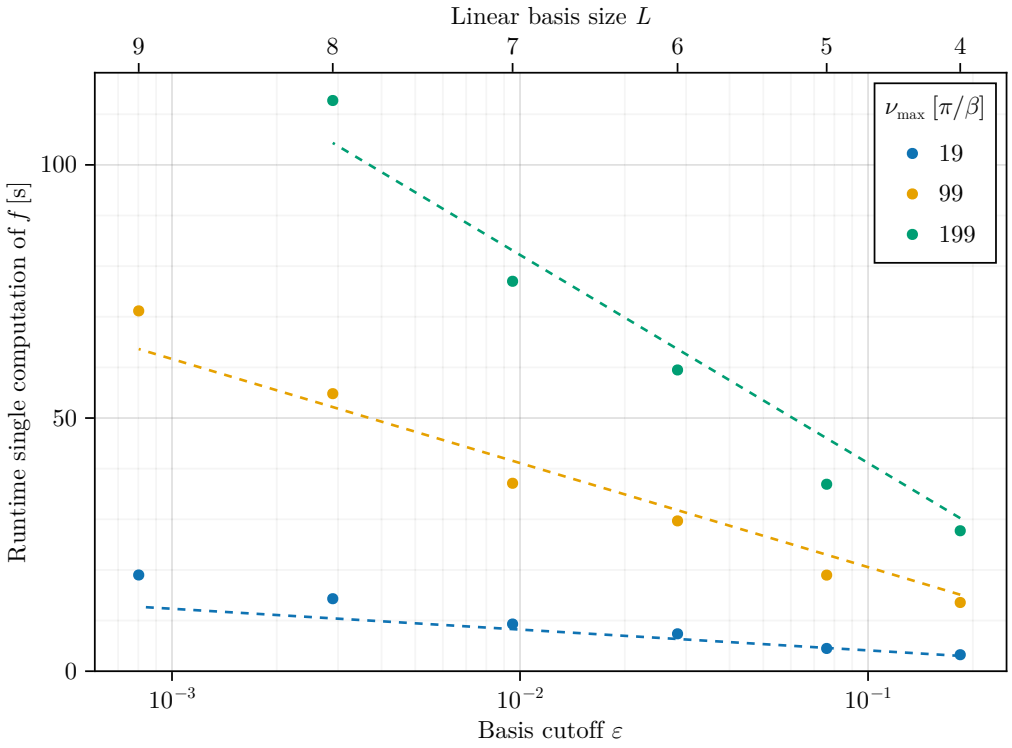

Figure 4.6: Time (in seconds) a single evaluation of f requires as a function of linear basis size L (top x-axis) and basis cutoff ε (bottom x-axis). For each point, 10 measurements were made and the minimum estimator taken [40]. The dashed lines indicate an $\mathcal{O}(\log(\varepsilon)\nu_{\max})$ model The times were measured running on 12 threads on an AMD Ryzen 9 7900X 12-Core Processor. The system considered is the half-filled Hubbard model with $U = 1t$ and $\beta = 5/t$ on 4×4 $\boldsymbol{k}$-points.

The time a single evaluation of f takes can be well approximated ($R^2 \approx 0.917$) by an $\mathcal{O}(\log(\varepsilon)\nu_{\max})$ model. One could have expected $\sim \mathcal{O}\left(\log(\varepsilon)^5\right)$ scaling here, which is roughly the behavior of the sparse sampling set's size.

Next, we would like to know how our actual problem parameters, chiefly temperature β^{-1} and resolution in $\boldsymbol{k}$-space, influence performance. In addition to runtime, we measure the code's memory footprint during problem setup, which is when the bulk of allocation happens. The measurements, displayed in fig. 4.7, are "missing" some points, because for these problem sizes, the available memory on the benchmarking machine (128 GB) was insufficient. This issue could be mitigated by using fewer threads, but this would come

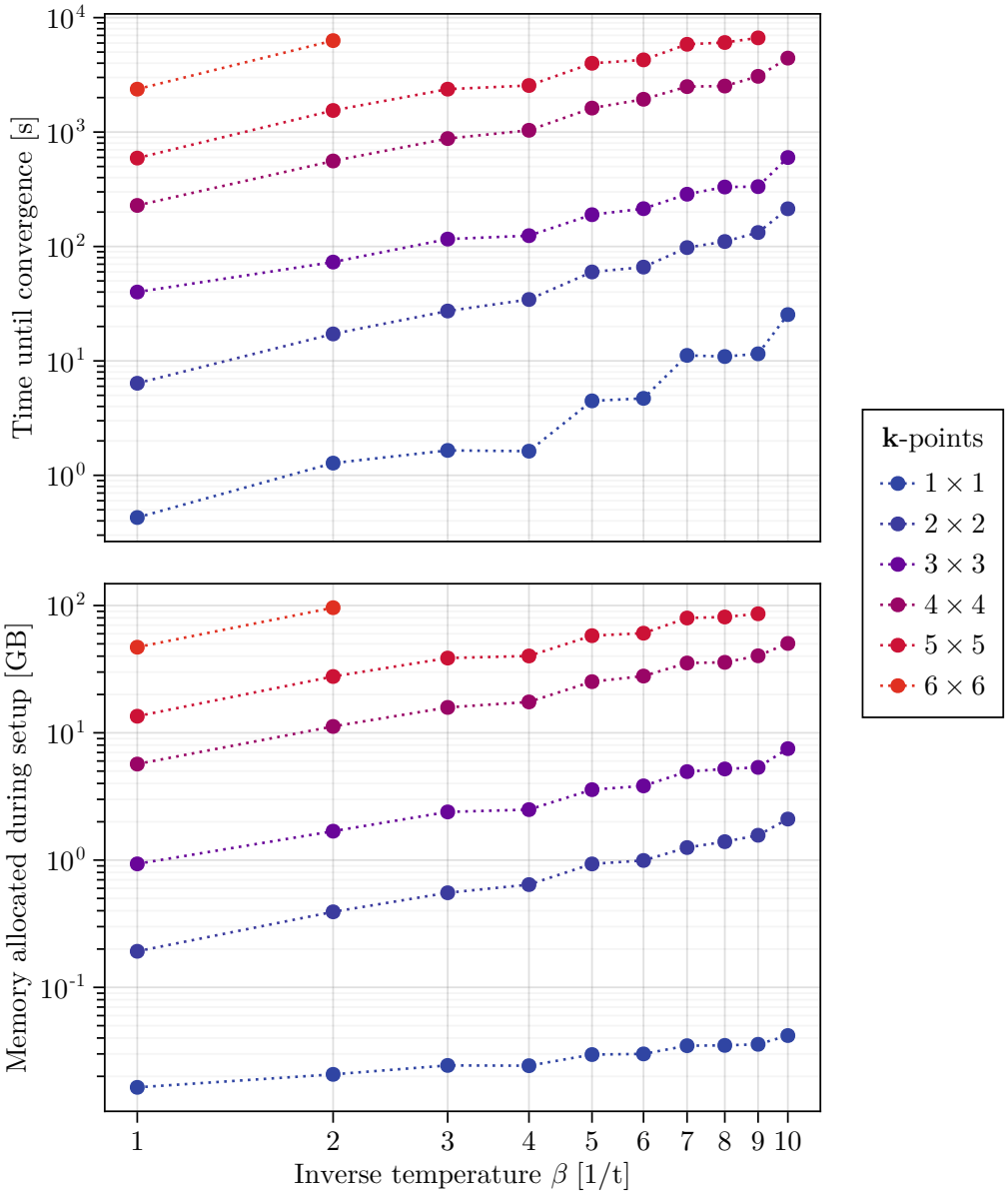

Figure 4.7: Time and space requirements of solving systems with decreasing temperature β^{-1} and increasing $\boldsymbol{k}$ resolution, with fixed interaction $U = 1t$. The times were measured running on 12 threads on an AMD Ryzen 9 7900X 12-Core Processor with 128 GB of memory. In all cases, $\varepsilon = 2^{-9}$ and $\nu_{\max} = 59\,\pi/\beta$. Note the abscissa's logarithmic scaling.

at the cost of speed. Furthermore, the resulting measurements would not be comparable to those shown here. Hence, we have chosen not to include them.

This benchmark indicates high costs both in time and space for increasing momentum space resolution. We also see here a feature that is particular to the way we perform symmetry reduction in momentum space: When going up a step from an "odd × odd" resolution there is a bigger jump in required computing resources than in stepping up from "even × even". Going back to fig. 3.2 elucidates this: even resolutions result in a higher number of points situated right on the boundary of the irreducible Brillouin zone which will be included in our symmetry-reduced bosonic momentum sampling set.

The scaling with β is more favorable in comparison.

4.3 Comparison with reference data

To verify the implementation's correctness, we compare to reference data computed by Anna Kauch with the MBE (Multi Boson Exchange) parquet solver [41, 42]. This yet unpublished box implementation boasts good (single boson exchange) treatment of asymptotics and, like the present work, computes one- and two-particle propagators.

For this comparison, we choose the half-filled Hubbard model at $U = 1t$, $\beta t \in \{2, 3, 4, 5\}$ on 4×4 $\boldsymbol{k}$-points on the square lattice with $t' = 0$. We then computed results at those same parameters using the discussed implementation with basis cutoff $\varepsilon = 10^{-3}$ and BSE sum cutoff $\nu_{\max} = 199\,\pi/\beta$ and present a comparison of the channel-reducible vertex Γ and of the self-energy Σ in figs. 4.8 to 4.11. Because there is a lot of data contained in $\Gamma^{r\nu\nu'\omega}_{\boldsymbol{k}\boldsymbol{k}'\boldsymbol{q}}$, we need to make some choices in presenting the results. Here we choose $\boldsymbol{k} = (\pi, 0)$, $\boldsymbol{k}' = (0, \pi/2)$, $\boldsymbol{q} = (\pi/2, \pi/2)$ and $\omega = 80\,\pi/\beta$. The self-energy $\Sigma^{\nu}_{\boldsymbol{k}}$ provides less difficulty in this regard, we choose $\boldsymbol{k} = (\pi/2, \pi)$. Note that points $\boldsymbol{k} = (k_x, k_y)$ with $|k_x - \pi| + |k_y - \pi| = \pi$ comprise the Fermi surface and are therefore special, for example, here $\varepsilon(\boldsymbol{k}) = 0$.

Straight away we see that agreement regarding Γ is excellent: The relative error on the presented subset of indices $\Delta_{\text{rel}} \equiv \|\Gamma_{\text{ref}} - \Gamma_{\text{own}}\|_\infty / \|\Gamma_{\text{ref}}\|_\infty$ is $\Delta_{\text{rel}} \approx 0.004$ at $\beta t = 2$ and increases linearly with β to $\Delta_{\text{rel}} \approx 0.009$ at $\beta t = 5$. Concerning Σ, the data disagree more: While qualitatively similar, the methods produce somewhat different figures at small frequencies ν. We again suspect the U^2 correction to be at fault here; investigations are ongoing at the time of writing.

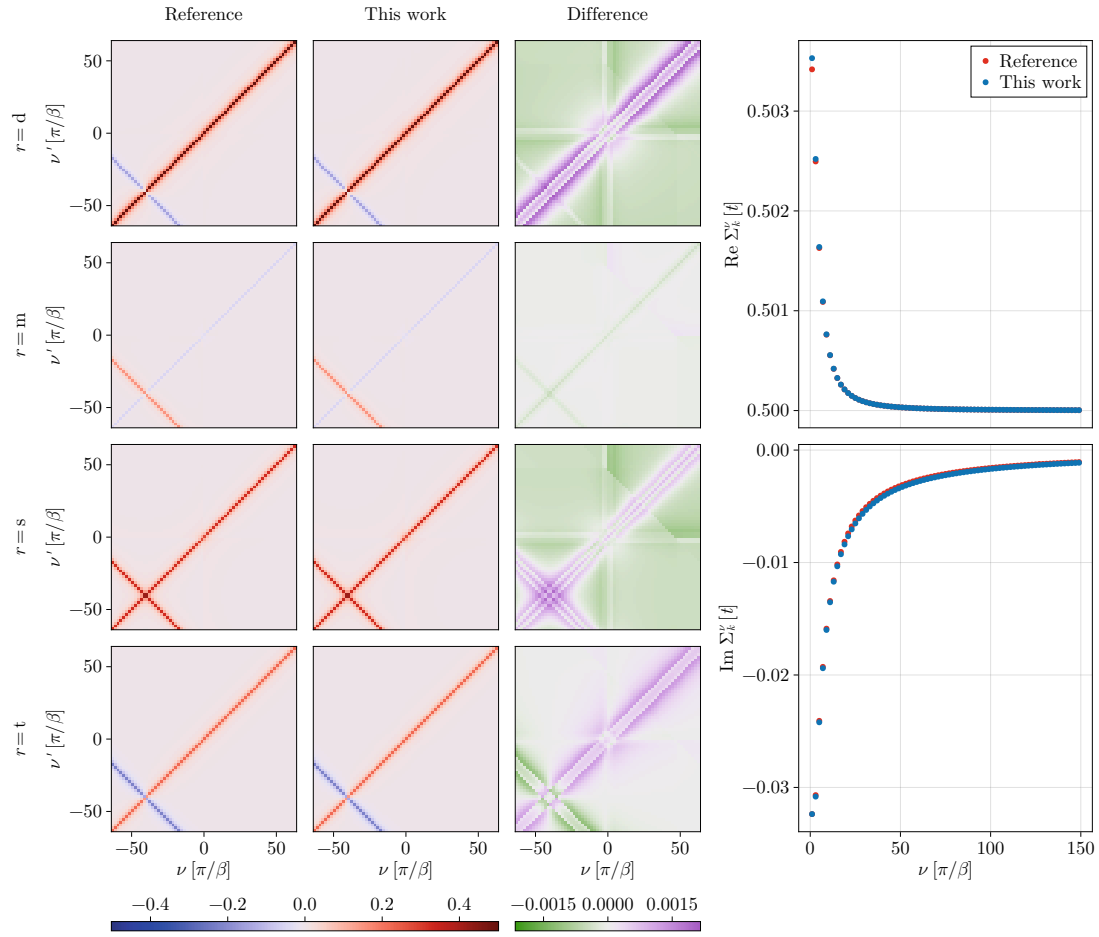

Figure 4.8: Comparison to reference data from the MBE parquet code. Shown is the channel-reducible Γ-vertex without background at $\omega = 80\,\pi/\beta$, $\boldsymbol{k} = (\pi, 0)$, $\boldsymbol{k}' = (0, \pi/2)$ and $\boldsymbol{q} = (\pi/2, \pi/2)$, (i) in the first column Γ_{own}, produced via the presented code, (ii) in the second column Γ_{ref} and (iii) in the third column $\Gamma_{\text{ref}} - \Gamma_{\text{own}}$. In the last column, the self-energy Σ at $\boldsymbol{k} = (\pi/2, \pi)$ is compared to reference data. Note that here, only the data on the sampling points are shown. In all cases the parquet approximation is used to approximate a solution to the $U = 1t$ and $\beta = 2/t$ half-filled Hubbard model on 4×4 $\boldsymbol{k}$-points.

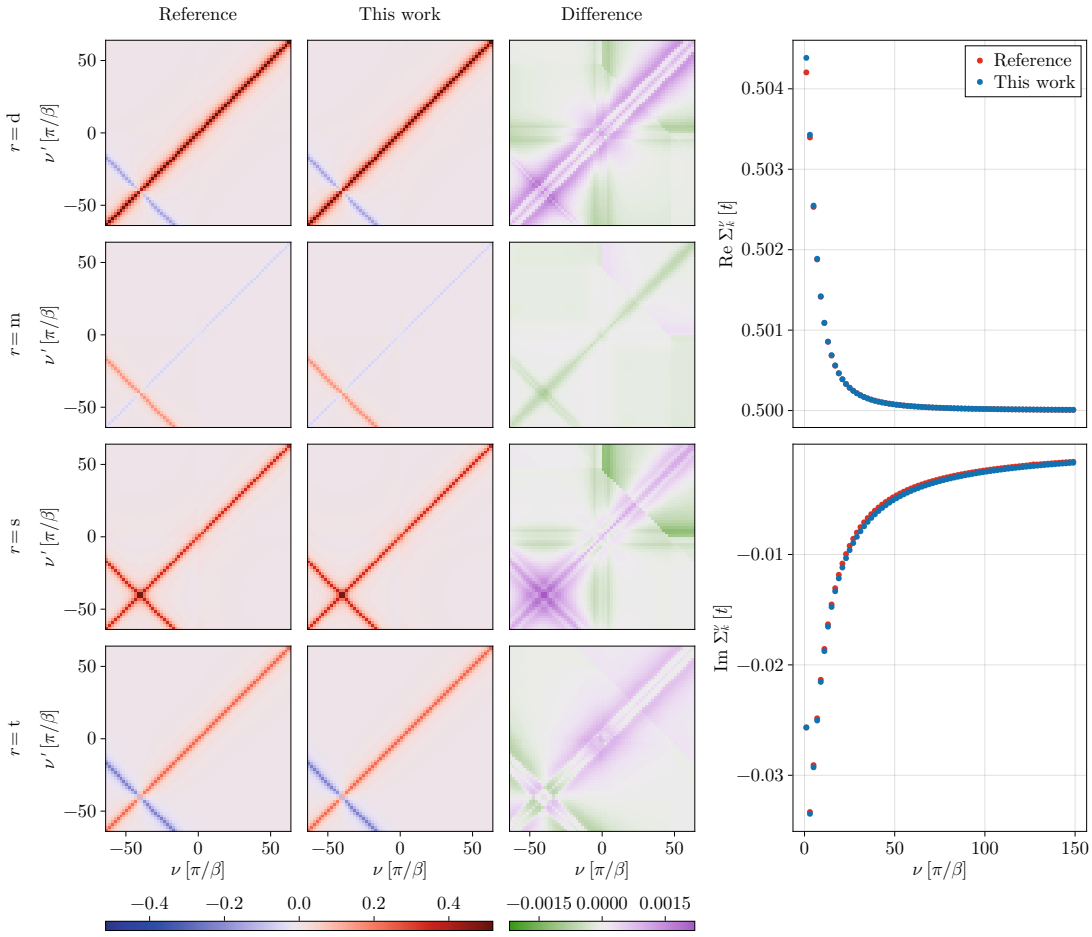

Figure 4.9: Same as fig. 4.8, but with $\beta = 3/t$.

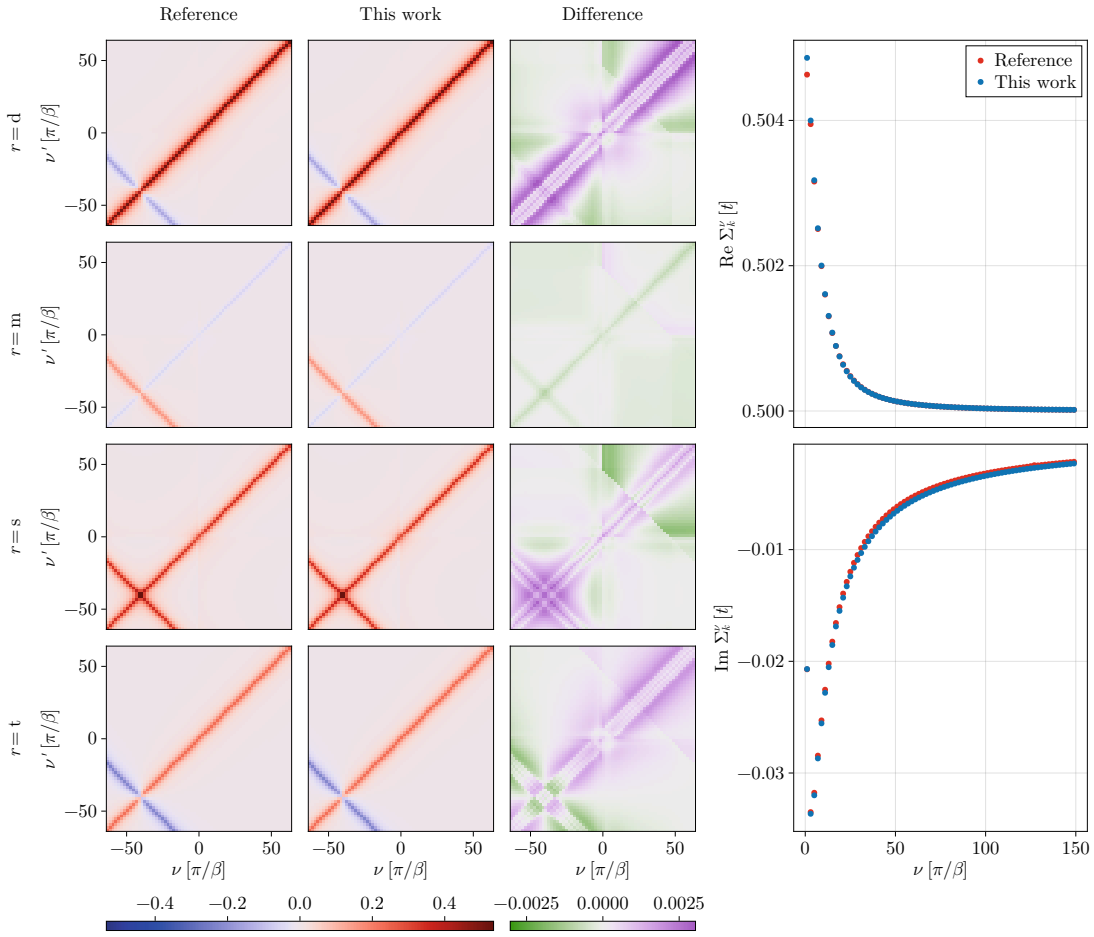

Figure 4.10: Same as fig. 4.8, but with $\beta = 4/t$.

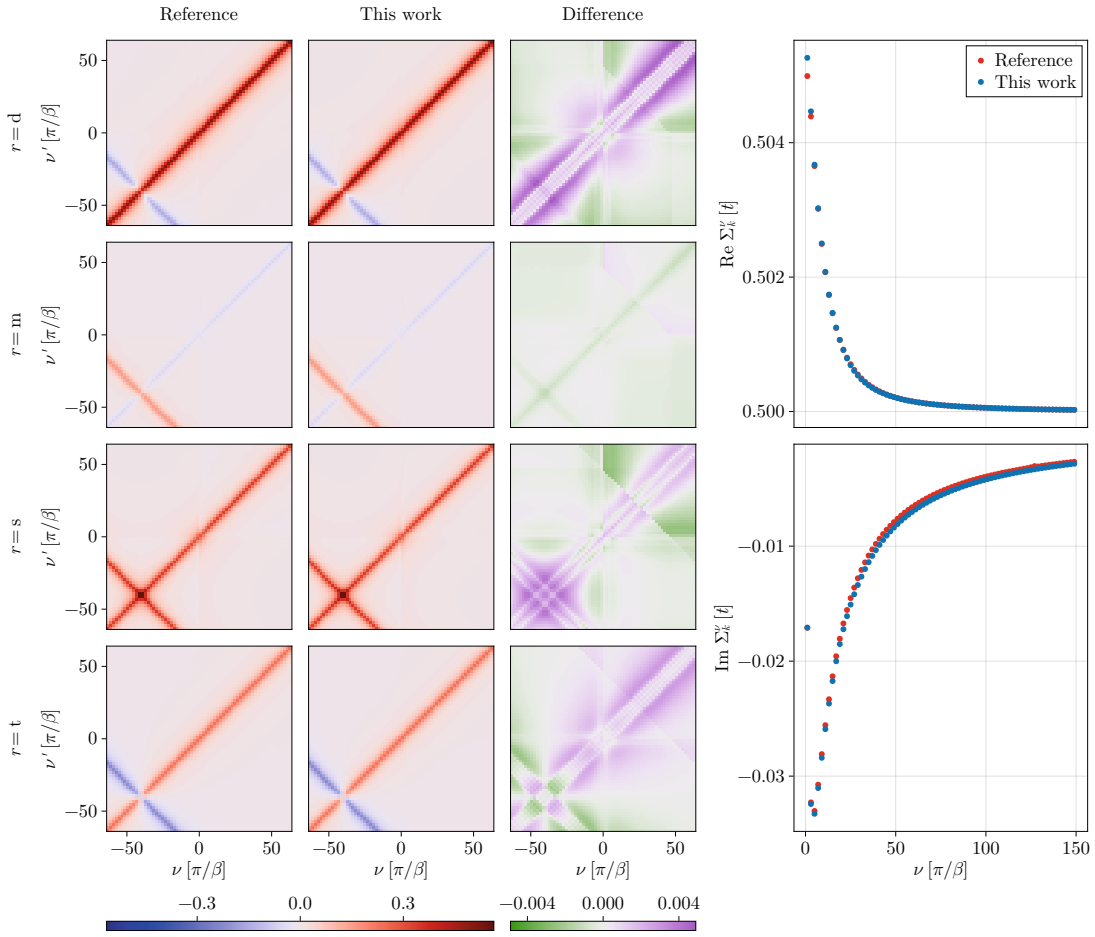

Figure 4.11: Same as fig. 4.8, but with $\beta = 5/t$.

4.4 Resulting vertices

Last, to try and convey some intuition for the vertices' structure, we would like to provide 3D visualizations of the results for half-filling, $U = 1t$, $\beta t = 5$ on a 4×4 $\boldsymbol{k}$-grid in figs. 4.12 to 4.14. Shown are the full vertex F, the channel-reducible vertex Φ and the self-energy Σ.

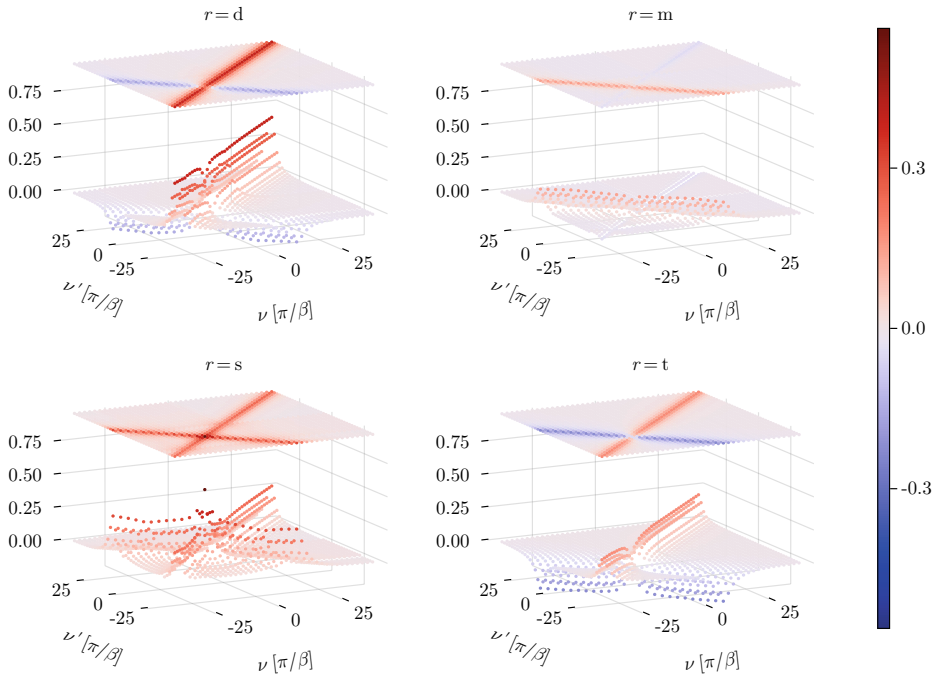

Figure 4.12: The half-filled Hubbard model in the parquet approximation with 4×4 $\boldsymbol{k}$-points at $U = 1t$ and $\beta = 5/t$. Full vertex with background subtracted $F^{r\nu\nu'\omega}_{\boldsymbol{k}\boldsymbol{k}'\boldsymbol{q}} - U^r$ with $\omega = 30\pi/\beta$ at $\boldsymbol{k} = (\pi/2, 0)$, $\boldsymbol{k}' = (\pi, 0)$ and $\boldsymbol{q} = (\pi, \pi)$. The vertex' values are shown once as a 3D scatter plot over the ν- and ν'-axes and once as a 2D heatmap floating above.

We can observe clearly that the assumption eq. (3.33a) fails to hold: The four-point vertices – after subtracting the background – rather than decaying at higher values of ν and ν' are full of diagonal planes running through them and extending to infinity. For example, F contains a plane at $\nu - \nu' \sim 0$ and another at $\nu + \nu' \sim -\omega$, both smeared out instead of sharp. This is a feature that is captured well by the overcomplete basis but invalidates any naive background removal shortcut.

Φ's structure is different from F's, featuring instead a "wedge" at

$$\begin{cases} -\omega \lesssim \nu \lesssim 0 & \text{if } \omega \geq 0 \\ 0 \lesssim \nu \lesssim -\omega & \text{if } \omega < 0 \end{cases} \tag{4.1}$$

and another described by substituting $\nu \to \nu'$ above. These too are fuzzy.

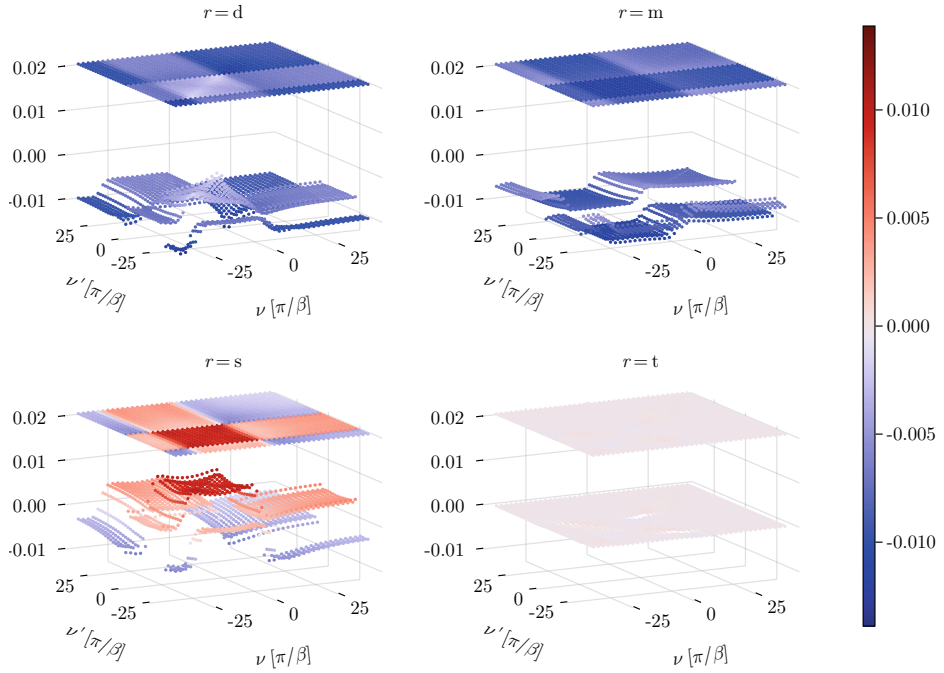

Figure 4.13: Same as fig. 4.12, but instead picturing the reducible vertex $\Phi^{r\nu\nu'\omega}_{\boldsymbol{k}\boldsymbol{k}'\boldsymbol{q}}$ with $\omega = 30\pi/\beta$ at $\boldsymbol{k} = (\pi/2, 0)$, $\boldsymbol{k}' = (\pi, 0)$ and $\boldsymbol{q} = (\pi, \pi)$.

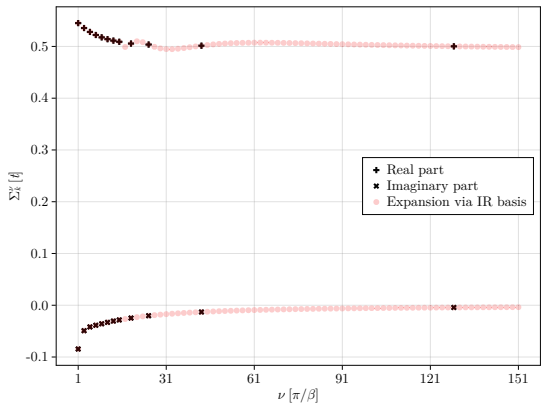

Figure 4.14: Same as fig. 4.12, but instead picturing the self energy $\Sigma(\nu)$ at $\boldsymbol{k} = (\pi, 0)$.

CHAPTER 5

Conclusion and outlook

In this thesis, we developed the first parquet solver using the IR-basis, advancing the state of the art in terms of performance and convergence properties.

It is implemented as a Julia library with unit tests covering all functions. We first applied this library to solve the Hubbard atom, where our solution agrees with the known exact solution. In his Bachelor's thesis [43] – which I had the opportunity to co-supervise – Matthias Michalek applied the code to the Anderson impurity model, with the results matching data produced by a reference solver.

Support for k-space dependence enabled us to solve the 2-dimensional Hubbard model with a 4×4 resolution on a desktop computer in a matter of minutes. Performance scaling and convergence properties were thoroughly tested and found to be favorable. Again, the results were compared to benchmark data and found to be in good agreement.

Still, much remains to be done. The multi-orbital parquet equations have been derived [23], so supporting multiple orbitals should be a straightforward extension. This new version would then presumably require more computing resources, i.e. an HPC cluster, so distributed memory parallelization would be a natural next step. Relatedly, optimization-wise there are still low-hanging fruits to be picked, such as the full Brillouin zone symmetry reduction briefly alluded to in a footnote in section 3.2.1. We should also explore more sophisticated techniques of storing the momentum dependence in the future. Quantics tensor trains [44, 45, 46] are a promising candidate for this. Also, the form-factor basis [47, 48] has already been applied and should be explored in our context as well.

The code discussed in this thesis is available at `https://github.com/tuwien-cms/ParquetIR.jl`. It supports Julia 1.9 and newer.

Bibliography

[1] A. Georges et al. „Dynamical mean-field theory of strongly correlated fermion systems and the limit of infinite dimensions“. In: *Reviews of Modern Physics* 68.1 (1996), pp. 13–125. DOI: 10.1103/RevModPhys.68.13.

[2] X. Blase et al. „The Bethe–Salpeter Equation Formalism: From Physics to Chemistry“. In: *Journal of Physical Chemistry Letters* 11.17 (2020), pp. 7371–7382. DOI: 10.1021/acs.jpclett.0c01875.

[3] G. Onida, L. Reining, and A. Rubio. „Electronic excitations: density-functional versus many-body Green's-function approaches“. In: *Reviews of Modern Physics* 74.2 (2002), pp. 601–659. DOI: 10.1103/RevModPhys.74.601.

[4] J. Lichtenstein et al. „Functional renormalization group study of an eight-band model for the iron arsenides“. In: *Physical Review B* 89.21 (2014), p. 214514. DOI: 10.1103/PhysRevB.89.214514.

[5] G. Esirgen and N. E. Bickers. „Fluctuation-exchange theory for general lattice Hamiltonians“. In: *Physical Review B* 55.4 (1997), pp. 2122–2143. DOI: 10.1103/PhysRevB.55.2122.

[6] Y. M. Vilk and A.-M. Tremblay. „Non-Perturbative Many-Body Approach to the Hubbard Model and Single-Particle Pseudogap“. In: *Journal de Physique I* 7.11 (1997), pp. 1309–1368. ISSN: 1286-4862. DOI: 10.1051/jp1:1997135.

[7] Y. Takada. „Inclusion of Vertex Corrections in the Self-Consistent Calculation of Quasiparticles in Metals“. In: *Physical Review Letters* 87.22 (2001), p. 226402. DOI: 10.1103/PhysRevLett.87.226402.

[8] E. Maggio and G. Kresse. „GW Vertex Corrected Calculations for Molecular Systems“. In: *Journal of Chemical Theory and Computation* 13.10 (2017), pp. 4765–4778. ISSN: 1549-9618. DOI: 10.1021/acs.jctc.7b00586.

[9] K. Zantout, S. Backes, and R. Valentí. „Effect of Nonlocal Correlations on the Electronic Structure of LiFeAs“. In: *Physical Review Letters* 123.25 (2019), p. 256401. DOI: 10.1103/PhysRevLett.123.256401.

[10] Y. Pavlyukh, G. Stefanucci, and R. van Leeuwen. „Dynamically screened vertex correction to GW“. In: *Physical Review B* 102.4 (2020), p. 045121. DOI: 10.1103/PhysRevB.102.045121.

[11] J. Kuneš. „Efficient treatment of two-particle vertices in dynamical mean-field theory“. In: *Physical Review B* 83.8 (2011), p. 085102. DOI: 10.1103/PhysRevB.83.085102.

[12] N. Lin, E. Gull, and A. J. Millis. „Two-Particle Response in Cluster Dynamical Mean-Field Theory: Formalism and Application to the Raman Response of High-Temperature Superconductors“. In: *Physical Review Letters* 109.10 (2012), p. 106401. DOI: 10.1103/PhysRevLett.109.106401.

[13] G. Rohringer, A. Valli, and A. Toschi. „Local electronic correlation at the two-particle level“. In: *Physical Review B* 86.12 (2012), p. 125114. DOI: 10.1103/PhysRevB.86.125114.

[14] H. Hafermann et al. „Collective charge excitations of strongly correlated electrons, vertex corrections, and gauge invariance“. In: *Physical Review B* 90.23 (2014), p. 235105. DOI: 10.1103/PhysRevB.90.235105.

[15] J. Otsuki, H. Hafermann, and A. I. Lichtenstein. „Superconductivity, antiferromagnetism, and phase separation in the two-dimensional Hubbard model: A dual-fermion approach“. In: *Physical Review B* 90.23 (2014), p. 235132. DOI: 10.1103/PhysRevB.90.235132.

[16] A. Galler et al. „Ab initio dynamical vertex approximation“. In: *Physical Review B* 95.11 (2017), p. 115107. DOI: 10.1103/PhysRevB.95.115107.

[17] A. Kauch et al. „Generic Optical Excitations of Correlated Systems: π-tons“. In: *Physical Review Letters* 124.4 (2020), p. 047401. DOI: 10.1103/PhysRevLett.124.047401.

[18] M. Kitatani et al. „Why the critical temperature of high-T_c cuprate superconductors is so low: The importance of the dynamical vertex structure“. In: *Physical Review B* 99.4 (2019), p. 041115. DOI: 10.1103/PhysRevB.99.041115.

[19] I. T. Diatlov, V. V. Sudakov, and K. A. Ter-Martirosian. „ASYMPTOTIC MESON-MESON SCATTERING THEORY“. In: *Soviet Phys. JETP* 5 (1957). URL: https://www.osti.gov/biblio/4338008.

[20] C. De Dominicis and P. C. Martin. „Stationary Entropy Principle and Renormalization in Normal and Superfluid Systems. II. Diagrammatic Formulation“. In: *Journal of Mathematical Physics* 5.1 (1964), pp. 31–59. DOI: 10.1063/1.1704064.

[21] J. Bezanson et al. „Julia: A fresh approach to numerical computing“. In: *SIAM review* 59.1 (2017), pp. 65–98. DOI: 10.1137/141000671.

[22] P. Thunström et al. „Analytical investigation of singularities in two-particle irreducible vertex functions of the Hubbard atom“. In: *Physical Review B* 98.23 (2018), p. 235107. DOI: 10.1103/PhysRevB.98.235107.

[23] J. Mangott. *Correlations in multi-orbital electronic systems: parquet equations*. Master thesis, TU Wien. 2022. DOI: 10.34726/hss.2022.101270.

[24] H. Shinaoka et al. „Compressing Green's function using intermediate representation between imaginary-time and real-frequency domains“. In: *Physical Review B* 96.3 (2017), p. 35147. ISSN: 2469-9969. DOI: 10.1103/PhysRevB.96.035147.

[25] J. Hubbard and B. H. Flowers. „Electron correlations in narrow energy bands“. In: *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences* 276.1365 (1963), pp. 238–257. DOI: 10.1098/rspa.1963.0204.

[26] N. E. Bickers. „Self-Consistent Many-Body Theory for Condensed Matter Systems“. In: *Theoretical Methods for Strongly Correlated Electrons. CRM Series in Mathematical Physics*. Ed. by D. Sénéchal, A.-M. Tremblay, and C. Bourbonnais. Springer, 2004. DOI: 10.1007/0-387-21717-7_6.

[27] G. Rohringer et al. „Diagrammatic routes to nonlocal correlations beyond dynamical mean field theory“. In: *Reviews of Modern Physics* 90.2 (2018), p. 025003. DOI: 10.1103/RevModPhys.90.025003.

[28] F. B. Kugler and J. von Delft. „Multiloop Functional Renormalization Group That Sums Up All Parquet Diagrams“. In: *Physical Review Letters* 120.5 (2018), p. 057403. DOI: 10.1103/PhysRevLett.120.057403.

[29] D. G. Anderson. „Iterative Procedures for Nonlinear Integral Equations“. In: *Journal of the ACM* 12.4 (1965), pp. 547–560. ISSN: 0004-5411. DOI: 10.1145/321296.321305.

[30] H. F. Walker and P. Ni. „Anderson Acceleration for Fixed-Point Iterations“. In: *SIAM Journal on Numerical Analysis* 49.4 (2011), pp. 1715–1735. DOI: 10.1137/10078356X.

[31] C. T. Kelley. *SIAMFANLEquations.jl*. Julia Package. 2022. DOI: 10.5281/zenodo.4284807. URL: https://github.com/ctkelley/SIAMFANLEquations.jl.

[32] H. J. Monkhorst and J. D. Pack. „Special points for Brillouin-zone integrations“. In: *Physical Review B* 13.12 (1976), pp. 5188–5192. DOI: 10.1103/PhysRevB.13.5188.

[33] N. W. Ashcroft and N. D. Mermin. *Solid state physics*. New York, NY: Holt, Rinehart and Winston, 1976. URL: https://cds.cern.ch/record/102652.

[34] J. A. C. Weideman. „Numerical Integration of Periodic Functions: A Few Examples“. In: *The American Mathematical Monthly* 109.1 (2002), pp. 21–36. ISSN: 00029890, 19300972. URL: http://www.jstor.org/stable/2695765 (visited on 05/06/2024).

[35] D. Cruz-Uribe and C. J. Neugebauer. „Sharp error bounds for the trapezoidal rule and Simpson's rule“. In: *J. Inequal. Pure Appl. Math* 3.4 (2002), pp. 1–22.

[36] M. Wallerberger et al. „sparse-ir: Optimal compression and sparse sampling of many-body propagators“. In: *SoftwareX* 21 (2023), p. 101266. ISSN: 2352-7110. DOI: 10.1016/j.softx.2022.101266.

[37] M. Wallerberger, H. Shinaoka, and A. Kauch. „Solving the Bethe-Salpeter equation with exponential convergence“. In: *Physical Review Research* 3.3 (2021), p. 033168. DOI: 10.1103/PhysRevResearch.3.033168.

[38] H. Shinaoka et al. „Overcomplete compact representation of two-particle Green's functions“. In: *Physical Review B* 97.20 (2018), p. 205111. ISSN: 2469-9969. DOI: 10.1103/PhysRevB.97.205111.

[39] S. Dirnböck et al. *Overcomplete intermediate representation of two-particle Green's functions and its relation to partial spectral functions*. 2024. DOI: 10.48550/arxiv.2404.05541. arXiv: 2404.05541 [cond-mat.str-el].

[40] J. Chen and J. Revels. „Robust benchmarking in noisy environments“. In: *arXiv preprint arXiv:1608.04295* (2016).

[41] F. Krien and A. Kauch. „The plain and simple parquet approximation: single-and multi-boson exchange in the two-dimensional Hubbard model“. In: *The European Physical Journal B* 95.4 (2022), p. 69. ISSN: 1434-6036. DOI: 10.1140/epjb/s10051-022-00329-6.

[42] F. Krien et al. „Boson-exchange parquet solver for dual fermions“. In: *Physical Review B* 102.19 (2020), p. 195131. DOI: 10.1103/PhysRevB.102.195131.

[43] M. Michalek. „Solving the Anderson impurity model with intermediate representation of the parquet equations“. 2024.

[44] H. Shinaoka et al. „Multiscale Space-Time Ansatz for Correlation Functions of Quantum Systems Based on Quantics Tensor Trains“. In: *Physical Review X* 13.2 (2023), p. 021015. DOI: 10.1103/PhysRevX.13.021015.

[45] M. K. Ritter et al. „Quantics Tensor Cross Interpolation for High-Resolution Parsimonious Representations of Multivariate Functions“. In: *Physical Review Letters* 132.5 (2024), p. 056501. ISSN: 1079-7114. DOI: 10.1103/PhysRevLett.132.056501.

[46] Y. Núñez Fernández et al. „Learning Feynman Diagrams with Tensor Trains“. In: *Physical Review X* 12.4 (2022), p. 041018. DOI: 10.1103/PhysRevX.12.041018.

[47] C. J. Eckhardt et al. „Truncated-unity parquet equations: Application to the repulsive Hubbard model“. In: *Physical Review B* 98.7 (2018), p. 075143. DOI: 10.1103/PhysRevB.98.075143.

[48] C. J. Eckhardt et al. „Truncated unity parquet solver“. In: *Physical Review B* 101.15 (2020), p. 155104. DOI: 10.1103/PhysRevB.101.155104.