



TECHNISCHE
UNIVERSITÄT
WIEN

BACHELORARBEIT

Efficient Evaluation of the Fourier Transform of Multipoint Correlation Functions

ausgeführt am Institut für Festkörperphysik
der Technischen Universität Wien

unter der Anleitung von
Dr. Anna Kauch
Dr. Markus Wallerberger
Univ.-Prof. Dr. Karsten Held

durch

Dipl.-Ing. Markus Reichel, BSc

Matrikelnummer 01529191

November 30, 2023

Markus Reichel

Erklärung zur Verfassung der Arbeit

Dipl.-Ing. Markus Reichel, BSc

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 30.11.2023

Markus Reichel

Kurzfassung

Diese Arbeit beschäftigt sich mit dem Transformieren der Green'schen Funktion von imaginärer Zeit in den Frequenzraum. Green'sche Funktionen und ihre Fourier-Transformation sind nützliche Entitäten, da sie direkt als physikalische Größen wie die Spektralfunktion oder die Suszeptibilität messbar sind. Komplexe Vielteilchenprobleme können mit Zwei-, Drei- und Mehrpunktfunktionen modelliert werden. Da die analytische Evaluierung im Frequenzraum im Vergleich zur Evaluierung in der Zeit sehr aufwendig ist, werden effiziente Methoden zur Fourier-Transformation benötigt. Einige numerische Fourier-Transformationsmethoden werden miteinander verglichen und anhand von Genauigkeit und Laufzeit beurteilt. Die Methoden wurden auf fermionische Zweipunkt-, fermionische Dreipunkt- und bosonische Dreipunktfunktionen für Hubbard und Anderson-Impurity Modelle angewandt. Hauptergebnisse dieser Arbeit beinhalten die Implementation der exakten Repräsentationen der genannten Korrelationsfunktionen im Frequenzraum, als auch einige numerische Methoden für die Fourier-Transformation. Vor allem bei Dreipunktfunktionen beginnen, wegen Zeitordnung und einer damit resultierenden Diskontinuität, simplerer Ansätze zu scheitern, allerdings bringt die sogenannte SimplexQuad-Methode genauere und schnellere Ergebnisse als die anderen Ansätze, in der Größenordnung von zwei Zehnerpotenzen.

Abstract

This bachelor thesis deals with transforming Green's functions from the imaginary time domain to the frequency domain. Green's functions and their Fourier transform are useful entities, due to being directly measurable as physical quantities like the spectral function or the susceptibility. Complex many-body interactions can be modelled using two-, three- and multi-point functions. Due to the analytic evaluation in frequency space being very costly in comparison to evaluation in time, efficient Fourier transform methods are needed. Several numerical Fourier transform methods are compared with one another and assessed in terms of accuracy and runtime. The methods were applied to fermionic two-point, fermionic three-point and bosonic three-point correlation functions for Hubbard and Anderson impurity models. The main results of this work include the implementation of the exact representations of the correlation functions in frequency space, as well as several numerical methods for the Fourier transform. Especially for three-point functions, due to time-ordering and a resulting discontinuity, simpler approaches begin to fail, but the so-called SimplexQuad method yields more accurate and faster results than the other approaches by two orders of magnitude.

Contents

1	Introduction	1
1.1	Matsubara Green's functions	1
1.2	Hubbard model	3
2	Methods	5
2.1	Analytical methods	5
2.2	Numerical methods	6
3	Two-Point Green's Function	7
3.1	Lehmann Representation	8
3.2	Analytic (Fermionic) Transformation	8
3.3	Naive Transformation	9
3.4	Gauss Quadrature	10
3.5	Intermediate Representation	10
3.6	Analysis	12
4	Three-Point Green's Function	17
4.1	Lehmann Representation	17
4.2	Analytic (Fermionic) Transformation	19
4.3	Naive Transformation	21
4.4	Gauss Quadrature	22
4.5	Analysis	23
4.6	Nonlinear (Bosonic) Response with exact Diagonalization	24
4.7	Analysis (Nonlinear Response)	28
5	Discussion	35
	Bibliography	i

1 Introduction

Correlation functions are important and useful entities in physics. The so-called propagator method [4, p. viii] can be used to derive essential properties of one- and two-particle quantum mechanics while being formulated in terms of quantities that can be directly observed in experiments. An example application is quantum chemistry [14, p. 60] where the Green's function method has the advantage that relevant physical properties like transition moments and ionization energies can be computed at firsthand in comparison to the wave function approach. For the latter, the ground state and the ionized states have to be calculated separately, which represents another source of error. Often, propagator functions are needed in frequency, but it is often faster to calculate them in time. This work looks at efficient ways of transforming Matsubara Green's functions from imaginary time into the frequency domain. Matsubara Green's functions are a handy tool to describe many-body interactions. They can be related to spectral measurements after analytic continuation, and their version in frequency space also links to physical observables. The Hamiltonian of the Hubbard model [1], and later the Anderson impurity model [11, p. 3-4], was used.

This work is structured as follows. First, we introduce Green's functions and the Hubbard model in this chapter. Next, the analytical as well as the numerical methods are described. After that, separate chapters for the two- and three-point functions introduce the analytic calculation of the Fourier transform as well as the numerical Fourier transform methods for Green's functions. Afterwards, they present the error analysis of these methods. The three-point function chapter is extended by approaches to calculating three-point bosonic correlation functions for the Anderson impurity model [11]. In the end, the findings as well as future work are discussed.

1.1 Matsubara Green's functions

Correlation functions or Green's functions contain information on the propagation of one or multiple particles [16]. The so-called causal Green's function

$$G_C(r^\theta, t^\theta, r, t) = i\hbar \hat{\psi}(r^\theta, t^\theta) \hat{\psi}^\dagger(r, t) i\theta(t^\theta - t) - i\hbar \hat{\psi}^\dagger(r, t) \hat{\psi}(r^\theta, t^\theta) i\theta(t - t^\theta) \quad (1.1)$$

describes the propagation of a particle from r to r^θ . The time dependence is represented in the Heisenberg picture. It assumes that a particle or hole is created at time t , propagates through the system and is then annihilated at time t^θ . $\hat{\psi}$ and $\hat{\psi}^y$ denote the annihilation and creation operators, θ the Heaviside function, and the plus sign is used for fermions and the minus for bosons. The angle brackets are used to calculate the expectation value regarding the temperature and the grand canonical ensemble and represent

$$\langle \hat{x} \rangle = \frac{\text{Tr } x e^{-\beta(\hat{H} - \mu\hat{N})}}{\text{Tr } e^{-\beta(\hat{H} - \mu\hat{N})}} \quad (1.2)$$

for $T > 0$, and

$$\langle \hat{x} \rangle = \langle jGS | x | jGS \rangle \quad (1.3)$$

for $T = 0$, with $|jGS\rangle$ being the ground state. Equation (1.2) can also be written as $\frac{1}{Z} \text{Tr } x e^{-\beta(\hat{H} - \mu\hat{N})}$ with the partition function $Z = \text{Tr } e^{-\beta(\hat{H} - \mu\hat{N})}$ and $\beta = \frac{1}{T}$. In Equation (1.1), the Heaviside functions θ perform the time ordering of t^θ and t , which can also be written with the Wick time ordering operator T :

$$G_C(r^\theta, t^\theta, r, t) = -i\hbar T \hat{\psi}(r^\theta, t^\theta) \hat{\psi}^y(r, t) \quad (1.4)$$

A useful variable transformation is the so-called Wick rotation, with maps $t \rightarrow i\tau$. This gives the so-called Matsubara or temperature Green's function

$$G(r^\theta, \tau^\theta, r, \tau) = \hbar T \hat{\psi}(r^\theta, i\tau^\theta) \hat{\psi}^y(r, i\tau) \quad (1.5)$$

Keep in mind that we are still in Heisenberg picture. In the Schrödinger picture, we get the following temperature Green's function (we look at the first term):

$$G(r^\theta, \tau^\theta, r, \tau) = \langle \hat{x} e^{+\hat{H}\tau^\theta} \hat{\psi}(r^\theta) e^{-\hat{H}\tau^\theta} e^{+\hat{H}\tau} \hat{\psi}^y(r) e^{-\hat{H}\tau} i\theta(t^\theta - t) \dots i\theta(t - t^\theta) \rangle \quad (1.6)$$

$$= \frac{1}{Z} \text{Tr} \left[e^{-\hat{H}(\beta - \tau^\theta)} \hat{\psi}(r^\theta) e^{-\hat{H}(\tau^\theta - \tau)} \hat{\psi}^y(r) e^{-\hat{H}\tau} \right] \theta(t^\theta - t) \dots i\theta(t - t^\theta) \quad (1.7)$$

where it can be seen that with the Wick rotation, β and τ remain both on the imaginary time axis. This gives the name of temperature Green's function and makes calculations of this function easier.

If the Hamiltonian is time-independent, the Green's function becomes invariant under time translations, which results in the Green's function being only dependent on the time difference.

$$G(r^\theta, \tau^\theta, r, \tau) = G(r^\theta, \tau^\theta - \tau, r, 0) \quad (1.8)$$

We from now on just use $\tau \in [0, \beta]$. One can show that $G(r^\ell, \tau, r, 0)$ is periodic or anti-periodic with period β . Therefore, we can define its Fourier transform as an integral evaluated between 0 and β .

$$\mathcal{G}(r^\ell, r, i\omega_n) = \int_0^\beta d\tau G(r^\ell, \tau, r, 0) e^{i\omega_n \tau} \quad (1.9)$$

And the original function in time space is retrieved again by an inverse transform.

$$G(r^\ell, r, \tau) = \sum_n \mathcal{G}(r^\ell, r, i\omega_n) e^{-i\omega_n \tau} \quad (1.10)$$

Also, this property of periodicity in time space results in a discrete number of frequencies in frequency space, explaining the sum in the inverse transformation Equation (1.10). The discrete frequencies ω_n are called Matsubara frequencies with these formulas:

$$\omega_n = \frac{2n\pi}{\beta} \quad \text{for bosons} \quad (1.11)$$

$$\omega_n = \frac{(2n+1)\pi}{\beta} \quad \text{for fermions.} \quad (1.12)$$

The two-point Green's function can be easily generalized to m -point Green's function with m operators. The Green's function in frequency space is a physically important entity; however, the Fourier transform from time to frequency can quickly get very hard. By using an equidistant grid with n_τ^ℓ points on each axis, the integral in Equation (1.9) has to be evaluated for every τ and every ω_n , which results in a transformation runtime of $O(n_\tau^\ell m \omega_n^m)$ for an m -point Green's function.

1.2 Hubbard model

In this work, we exemplify the Fourier transform of the correlation functions for the Hubbard model. The Hubbard model [1] is a strongly simplified concept of modelling the atoms in a solid as sites and the electrons having the ability to hop between those sites. Double-occupied sites are punished with energy cost U , which corresponds to the screened Coulomb repulsion. Spin- $\frac{1}{2}$ electrons interact and hop on a set of L sites, which are spatially localized orbitals. The Hamiltonian is

$$\hat{H} = \sum_{i,j \in \mathcal{L}} \sum_{\sigma} t_{ij} \hat{c}_{i\sigma}^\dagger \hat{c}_{j\sigma} + U \sum_{i \in \mathcal{L}} \hat{c}_{i\uparrow}^\dagger \hat{c}_{i\downarrow}^\dagger \hat{c}_{i\downarrow} \hat{c}_{i\uparrow} \quad (1.13)$$

with $\hat{c}_{i\sigma}^{(\dagger)}$ being annihilation(creation) operator for an electron with spin σ on site i . The hopping integral t_{ij} between sites is limited to the nearest-neighbours in our

case and equal to t . Therefore, the first term [16, p. 4] represents a kinetic energy term, while the second term equals a potential energy term with the Coulomb integral U . It describes the repulsion between electrons on the same site. $\hat{c}_{i\sigma}^\dagger \hat{c}_{i\sigma}$ equals $\hat{n}_{i\sigma}$, the number operator on the site i with spin σ . In the course of this work, the Hamiltonian is described together with the chemical potential μ by defining a new Hamiltonian in order to write \hat{H} instead of $\hat{H} - \mu \hat{N}$ (we work in grand canonical ensemble) :

$$\hat{H} - \mu \hat{N} \equiv \hat{H} \quad (1.14)$$

This results in

$$\hat{H} = -t \sum_{i,j} \sum_{\sigma} \hat{c}_{i\sigma}^\dagger \hat{c}_{j\sigma} + U \sum_{i,j} \hat{n}_i \hat{n}_j - \mu \sum_{i,j} (\hat{n}_i + \hat{n}_j) \quad (1.15)$$

In particular, we will use the two-site Hubbard Hamiltonian ($d = f1, 2g$):

$$\begin{aligned} \hat{H}_2 = & -t(\hat{c}_{1\sigma}^\dagger \hat{c}_{2\sigma} + \hat{c}_{1\sigma} \hat{c}_{2\sigma}^\dagger + \hat{c}_{2\sigma}^\dagger \hat{c}_{1\sigma} + \hat{c}_{2\sigma} \hat{c}_{1\sigma}^\dagger) \\ & + U(\hat{n}_{1\sigma} \hat{n}_{1\sigma} + \hat{n}_{2\sigma} \hat{n}_{2\sigma}) - \mu(\hat{n}_{1\sigma} + \hat{n}_{1\sigma} + \hat{n}_{2\sigma} + \hat{n}_{2\sigma}) \end{aligned} \quad (1.16)$$

The one-site Hubbard Hamiltonian ($d = f1g$) has no hopping terms t between sites.

$$\hat{H}_1 = U \hat{n}_1 \hat{n}_1 - \mu(\hat{n}_1 + \hat{n}_1) \quad (1.17)$$

2 Methods

An in-depth theoretical foundation of the propagator or Green's function method is given in the book by Dickhoff and Van Neck [4], where basics like the second quantization, statistical mechanics and diagram rules, but also advanced topics beyond the mean field approximation are discussed. Mattuck [13] also gives an overview of propagators, with more focus on intuition and Feynman diagrams. In order to find and compare different ways to efficiently evaluate the Fourier transform of the multipoint correlation functions, both the two-point (one particle) as well as the three-point Green's functions were transformed from the time domain to Matsubara frequencies. Different methods found in the literature and the internet were used and then compared to the exact solution via error analysis.

2.1 Analytical methods

The Lehmann representation of the correlation functions was used as the exact method to benchmark the correctness of the other numeric approaches. It is possible to find the exact solution of the Hubbard model for several lattice sites with the exact diagonalization method. It was feasible to compute the two- and three-point Green's functions for the two-site Hubbard model and calculate explicitly the Lehmann representation in time and frequency. However, the Lehmann representation of the three-point Green's function for the three-site bath Anderson impurity model (4 sites in total) was already pushing the device of the author (9 year old laptop HP Envy 17-K102NG with upgraded Samsung 870 EVO SATA III SSD) to its limits. To get the frequency domain representation of the Green's functions, they were analytically Fourier-transformed, first for the two- and the three-point function for fermions, and additionally later for the three-point function for bosons. Additional calculation for bosons was necessary because in this case, additional terms appear where the integration yields different functions. The reason is constant values in the integrand which come from the fact that bosonic frequencies can be zero.

2.2 Numerical methods

The computational part of this work was done in the programming language Julia and the Jupyter platform [10]. Fermion calculations were done using the Fermions.jl library by Markus Wallerberger [26], some calculations of the Green's function in time space as well as the Hubbard Hamiltonian generation were done using the project work of Stefan Rohshap [17].

Initially, a naive integration via a finite sum without weights was used. As already said, the analytical Lehmann representation was derived for comparison. Gauss quadrature algorithms were then probed in order to get a faster run-time. The quadrature algorithms used were Gauss-Legendre [6], Gauss Kronrod [15] (2-point) and SimplexQuad [22], Grundmann-Möller [9] (3-point). Due to the time ordering operator, in case of three-point Green's functions, the integration area is split into two simplices with the function values being non-continuous. Hence, it was necessary to use simplex integration algorithms and split the integration into two operations for each simplex.

In the end, for the two-point Green's function, the concept of intermediate representations was used as an approach different to the others, not by directly computing the transformation integral but by using a representation of the function between time and frequency space, therefore called intermediate representation (IR) [23]. The basics of the concept [19] [21] [20] are briefly discussed in Chapter 3.5. During the time of working on this thesis, the IR approach was not yet ready for the three-point Green's function. Because of this, the three-point Green's function chapter was extended with a nonlinear response of a particle bath system in Chapter 4.6, where the Lehmann representation as well as the Gauss quadrature was adapted for bosonic three-point Green's functions [11].

3 Two-Point Green's Function

We will now compute the two-point or one-particle Green's function for the Hubbard model, and then look at multiple approaches to get the Fourier-transformed function in frequency space. We set the temperature β and the operators \hat{A} and \hat{B} and use a function from Fermions.jl which computes $\hbar T \hat{A}(\tau) \hat{B}(0) i$ at temperature $\frac{1}{\beta}$. Functions in Julia in imaginary time are already available. The exact computation for a small number of sites can be done using the Lehmann representation. To get the analytic transformation, we will first derive the Lehmann representation in imaginary time, and then calculate the Fourier transform.

The practical computations are done as an example on a two-site Hubbard model (see Equation (1.16)) with $U = 3$, $t = 1$ and half-filling with $\mu = 1.5$. The inverse temperature β is 10. Operator \hat{A} is the creation operator $\hat{c}_{1\uparrow}^\dagger$, and operator \hat{B} is the annihilation operator $\hat{c}_{1\uparrow}$ (both on site $i = 1$ with spin up). With the Lehmann representation discussed in the next section, this gives us the $G_{AB}(\tau)$ in Figure 3.1.

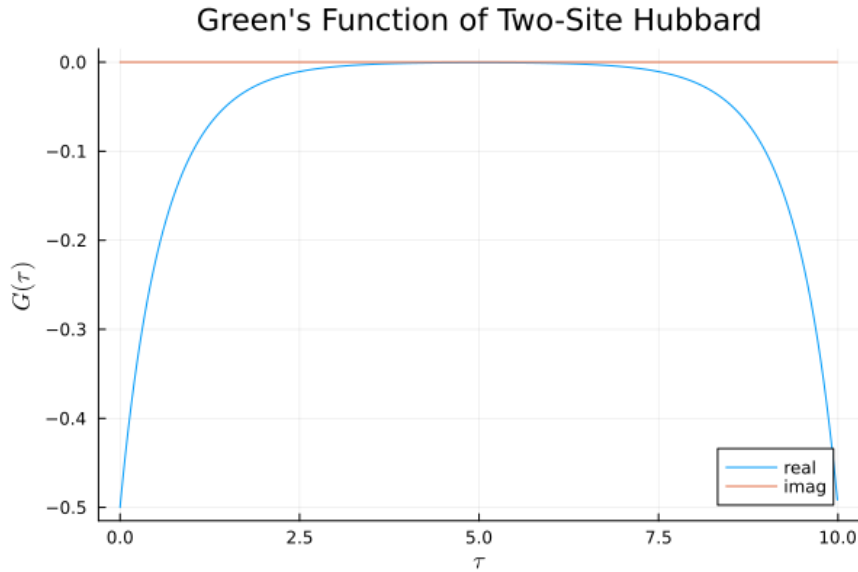


Figure 3.1: Real and imaginary part of the exact local Green's function (see Equation (3.4)) of the two-site Hubbard model for $U = 3$, $\beta = 10$, $t = 1$ and $\mu = 1.5$. The imaginary part is zero.

3.1 Lehmann Representation

The Lehmann representation of the two-point Green's function can be obtained as follows. The definition of G with two operators and one time is

$$G_{AB}(\tau) = \hbar T \hat{A}(\tau) \hat{B}(0) i \quad (3.1)$$

with the operators in the Heisenberg picture and $0 \leq \tau \leq \beta$. Converting Equation (3.1) into the Schrödinger picture yields

$$G_{AB}(\tau) = \hbar e^{\tau \hat{H}} \hat{A} e^{-\tau \hat{H}} \hat{B} i = \frac{1}{Z} \text{Tr} (e^{-(\beta - \tau) \hat{H}} \hat{A} e^{-\tau \hat{H}} \hat{B}) = \frac{1}{Z} \sum_n \langle m_j | e^{-(\beta - \tau) \epsilon_n} \hat{A} e^{-\tau \hat{H}} \hat{B} | n_i \rangle \quad (3.2)$$

By inserting the basis of eigenstates of \hat{H} , $|j m_i\rangle \langle m_j|$, in the last term of Equation (3.2), we get the Lehmann representation.

$$\frac{1}{Z} \sum_{n,m} \langle m_j | e^{-(\beta - \tau) \epsilon_n} \hat{A} e^{-\tau \hat{H}} | j m_i \rangle \langle m_j | \hat{B} | n_i \rangle = \frac{1}{Z} \sum_n e^{-\beta \epsilon_n} \sum_m e^{-\tau(\epsilon_m - \epsilon_n)} \langle m_j | \hat{A} | j m_i \rangle \langle m_j | \hat{B} | n_i \rangle \quad (3.3)$$

So $G_{AB}(\tau)$ can be written as

$$G_{AB}(\tau) = \frac{1}{Z} \sum_n G_{AB}^{(n)}(\tau) \quad (3.4)$$

with

$$G_{AB}^{(n)}(\tau) = e^{-\beta \epsilon_n} \sum_m e^{-\tau(\epsilon_m - \epsilon_n)} \langle m_j | \hat{A} | j m_i \rangle \langle m_j | \hat{B} | n_i \rangle \quad (3.5)$$

3.2 Analytic (Fermionic) Transformation

We now want to transform (3.5) into the frequency domain. Analytically, we write for the Fourier transform

$$\begin{aligned} G_{AB}(\omega) &= \int_0^\beta d\tau e^{i\omega\tau} G_{AB}(\tau) \\ &= \frac{1}{Z} \int_0^\beta d\tau e^{i\omega\tau} \sum_n e^{-\beta \epsilon_n} \sum_m e^{-\tau(\epsilon_m - \epsilon_n)} \langle m_j | \hat{A} | j m_i \rangle \langle m_j | \hat{B} | n_i \rangle \end{aligned} \quad (3.6)$$

$$= \frac{1}{Z} \int_0^\beta d\tau e^{i\omega\tau} \sum_{n,m} e^{-\beta \epsilon_n} e^{-\tau(\epsilon_m - \epsilon_n)} \langle m_j | \hat{A} | j m_i \rangle \langle m_j | \hat{B} | n_i \rangle \quad (3.7)$$

$$= \frac{1}{Z} \sum_{n,m} e^{-\beta \epsilon_n} \underbrace{\int_0^\beta d\tau e^{i\omega\tau} e^{-\tau(\epsilon_m - \epsilon_n)}}_{\text{We will evaluate this integral.}} \langle m | A | j \rangle \langle m | B | j \rangle \quad (3.8)$$

We evaluate the integral in Equation (3.8).

$$\int_0^\beta d\tau e^{i\omega\tau} e^{-\tau(\epsilon_m - \epsilon_n)} \quad (3.9)$$

$$= \int_0^\beta d\tau e^{-\tau(\epsilon_m - \epsilon_n - i\omega)} \quad (3.10)$$

$$= \frac{1}{\epsilon_m - \epsilon_n - i\omega} [e^{-\beta(\epsilon_m - \epsilon_n - i\omega)} - 1] \quad (3.11)$$

This gives us the Fourier-transformed Green's function in Lehmann representation:

$$G_{AB}(\omega) = \frac{1}{Z} \sum_{n,m} e^{-\beta \epsilon_n} \frac{1}{\epsilon_m - \epsilon_n - i\omega} [e^{-\beta(\epsilon_m - \epsilon_n - i\omega)} - 1] \langle m | A | j \rangle \langle m | B | j \rangle \quad (3.12)$$

When operator \hat{A} is fermionic, $\omega_n = \frac{(2n+1)}{\beta}\pi$, which means that $e^{\beta i\omega_n} = e^{i(2n+1)\pi} = -1$, which can be used in the calculation of Equation (3.12). The analytically transformed propagator of our two-site Hubbard Hamiltonian is plotted in Figure 3.2. Note that in (3.12) we have a double sum over all eigenstates, which makes the computation slow for larger systems.

3.3 Naive Transformation

The first approach to get faster run-time is quite naive. To get to the frequency domain, the Fourier integral is numerically approximated as a sum. For this, the imaginary time τ is sliced and inputted as an array, and then, the terms are summed up. This is done in the code via matrix multiplication. The computation is done as follows:

$$G(i\omega_n) = \sum_j e^{i\omega_n \tau_j} G(\tau_j) \quad (3.13)$$

By splitting τ into 1000 intervals, it is possible to get a Fourier transform for two-point functions with this simple approach. We will later see that beginning with the three-point function, due to time ordering and a resulting discontinuity, the naive approach begins to fail more quickly than in the two-point function case.

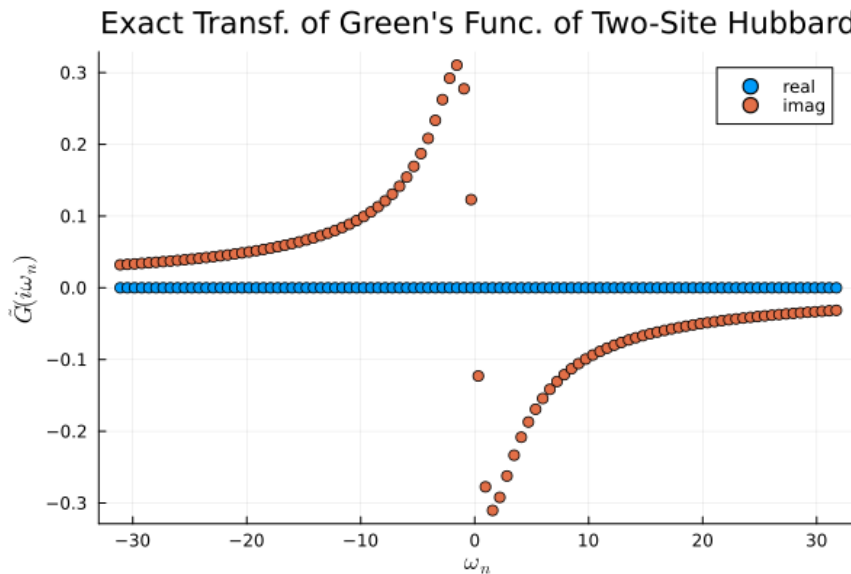


Figure 3.2: Real and imaginary part of the analytically transformed exact Green’s function (see Equation (3.12)) of the two-site Hubbard model with the operators and parameters as in Figure 3.1. Note that the Matsubara frequencies are discrete. The real part is zero.

3.4 Gauss Quadrature

The naive integration approach has the disadvantage of weighting every sub-interval the same. Therefore, several Gauss quadrature algorithms exist. They are similar to the naive transformation, but each summation term has an additional weight factor w_j . The two Julia packages `FastGaussQuadrature.jl` and `QuadGK.jl` are used to compute classic Gauss-Legendre and adaptive Gauss-Kronrod quadratures. For details on how the weights are calculated, a reference is made to the literature [3] [12].

$$G(i\omega_n) = \sum_j w_j e^{i\omega_n \tau_j} G(\tau_j) \quad \tau \quad (3.14)$$

3.5 Intermediate Representation

The idea of the intermediate representation (IR) is to store the Green’s function in a compact form between imaginary time and real frequency space in order to save computational cost and storage size [19]. The principle of IR is the fact that through analytic continuation, both parameter spaces share the so-called IR basis,

from where each representation can be reconstructed.

$$G(\tau_j) \text{ and } G(i\omega_n) \text{ share } g_l \quad (3.15)$$

The shared coefficients g_l can be derived through the fact that the Green's function can be written in terms of an integral kernel K and a spectral function ρ with the cutoff frequency ω_{max} :

$$G(\tau) = \int_{\omega_{max}}^{\omega_{max}} d\omega K(\tau, \omega) \rho(\omega) \quad (3.16)$$

The kernel and spectral function have different forms regarding the particle statistics (bosons or fermions, see Shinaoka et al. [21] for more details). If τ and ω are discrete, Equation (3.16) becomes the linear equation

$$G = K\rho \quad (3.17)$$

Now, the kernel K can be decomposed into two different orthogonal basis sets $fU_l g$ and $fV_l g$.

$$K(\tau, \omega) = \sum_{l=0}^{\gamma} s_l U_l(\tau) V_l(\omega) \quad (3.18)$$

In the discrete case, this corresponds to the singular value decomposition (SVD) of the kernel matrix:

$$K = USV^T \quad (3.19)$$

Let us now write the Green's function by using the complete basis U and the coefficients g_l we want to derive:

$$G(\tau) = \sum_{l=0}^{\gamma} g_l U_l(\tau) \quad (3.20)$$

If we now insert Equation (3.18) into Equation (3.16), set the variables which do not depend on ω to the front and compare it to Equation (3.20):

$$G(\tau) = \sum_{l=0}^{\gamma} s_l U_l(\tau) \underbrace{\int_{\omega_{max}}^{\omega_{max}} d\omega V_l(\omega) \rho(\omega)}_{\rho_l} \quad (3.21)$$

We can find a term for describing the g_l coefficients, namely

$$g_l = s_l \rho_l \quad (3.22)$$

with the ρ_l coefficients computed by the integral. Shinaoka et al. state that the s_l and therefore the g_l coefficients decay very fast, therefore, only a few basis

functions (small l) are needed in the sum of Equation (3.20) to reconstruct $G(\tau)$ from the g_l coefficients and the U_l basis reasonable accurately. Equation (3.22) tells us how we can get the g_l coefficients from the SVD, but it is also valid to calculate them (with discrete τ_j) as a minimalization problem:

$$g_l = \arg \min_g \sum_j |G(\tau_j) - \sum_l g_l U_l(\tau_j)|^2 \quad (3.23)$$

This will be used by us in our calculation. But how is $\mathcal{G}(i\omega)$ calculated from the same g_l coefficients? Fourier-transforming $G(\tau)$ with its representation in (3.20) gives the answer:

$$\begin{aligned} \mathcal{G}(i\omega_n) &= \int_0^\beta d\tau e^{i\omega_n \tau} G(\tau) = \int_0^\beta d\tau e^{i\omega_n \tau} \sum_{l=0}^1 g_l U_l(\tau) \\ &= \sum_{l=0}^1 g_l \underbrace{\int_0^\beta d\tau e^{i\omega_n \tau} U_l(\tau)}_{\mathcal{U}_l(i\omega_n)} = \sum_{l=0}^1 g_l \mathcal{U}_l(i\omega_n) \end{aligned} \quad (3.24)$$

We can therefore describe $G(\tau)$ and $\mathcal{G}(i\omega_n)$ with the same g_l and the $U_l(\tau) / \mathcal{U}_l(i\omega_n)$ basis functions.

3.6 Analysis

We now compare the four ways of Fourier-transforming the Green's function (naive, adaptive Gauss-Kronrod, Gauss-Legendre, IR) with the exact Lehmann representation. First, we calculate the difference between the results of these methods and the Lehmann result in a log plot. The number of function evaluation points N is comparable. Figure 3.3 shows this plot. We can see that the naive method does not get even close to high accuracy. The Gauss methods become very accurate in $\omega_n \in [-10, 10]$. The IR method is accurate on the whole omega range for this number of function evaluation points. Next, the error of these methods is compared in a log plot for different numbers of points. To compute the error, the maximum norm was used:

$$\text{Error} = \max_j |G(i\omega_n) - \mathcal{G}(i\omega_n)| \quad (3.25)$$

Due to the real part vanishing, we will use the imaginary part for this error measure. For the naive and Gauss-Legendre approach, the number of input points will also be the number of points used. In case of Gauss-Kronrod, due to it being adaptive, the average number of points was used as a benchmark tool. In case of IR, one can prove that the number of basis functions used represents the number of

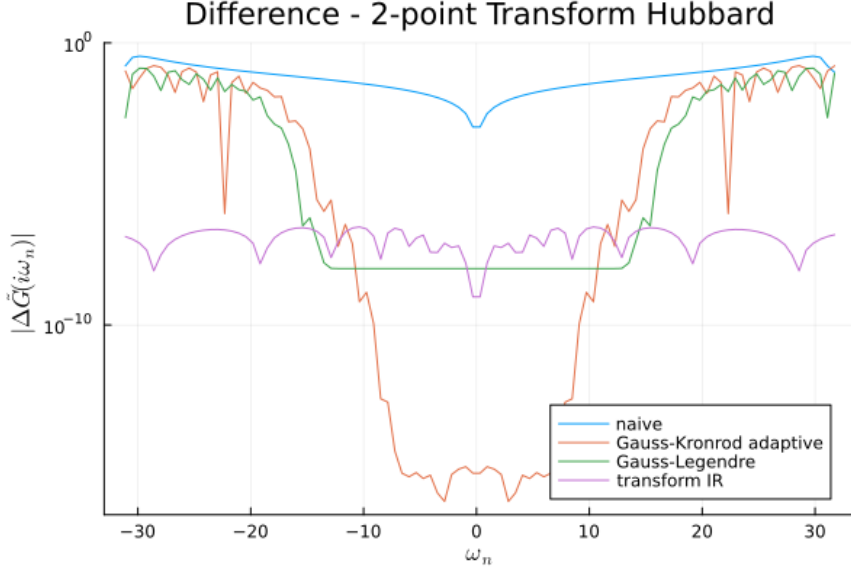


Figure 3.3: Absolute difference of the imaginary part of $\mathcal{G}(i\omega_n)$ obtained with various methods (see box) to the exact Lehmann transformation. The number of τ sampling points used in this case is comparable with $N_{naive} = 50$, $N_{Kronrod} = 42.6$, $N_{Legendre} = 45$ and $N_{IR} = 49$. For the computational effort of the methods, see the next plots below.

points (therefore, $L = N$). Figure 3.4 shows the log plot of the error as a function of the number of corresponding sampling points. Note that initially in this work, the function to compute the Green's function $G(\tau)$ had some noise in it:

$$G(\tau)^{w/ \text{noise}} = G(\tau) + \epsilon_N \quad (3.26)$$

With the noise $\epsilon_N \approx 10^{-10}$. Due to this noise term, the IR basis could not completely reconstruct $G(\tau)^{w/ \text{noise}}$, because the initial assumption was $G(\tau) = \sum_{l=0}^{L-1} g_l U_l(\tau)$. The IR reconstruction can also be seen as the projector $P(\tau, \tau^\theta) = \sum_{l=0}^{L-1} U_l(\tau) U_l(\tau^\theta)$, and while applying $P(\tau, \tau^\theta)$ to $G(\tau)$ gives $\sum_{l=0}^{L-1} g_l U_l(\tau^\theta)$ and therefore $G(\tau^\theta)$, applying the projector to Equation 3.26 yields the additional error term $\sum_{l=0}^{L-1} U_l(\tau) U_l(\tau^\theta) \epsilon_N$. This is shown in Figure 3.4 with "transform IR with ϵ_N ", where the error stays the same around 10^{-10} even for higher N . After this issue was detected, a better $G(\tau)$ function with at least $\epsilon_N \approx 2 \cdot 10^{-16}$ (about machine precision) was used, which resulted in the error curve "transform IR without noise". Note that in this case, the noiseless IR converges with the least number of points (resp. basis functions). From now on, in the following plots, only this better function was used. A very important relationship, namely the absolute error as a function of runtime, can be seen in Figure 3.5. Interestingly enough,

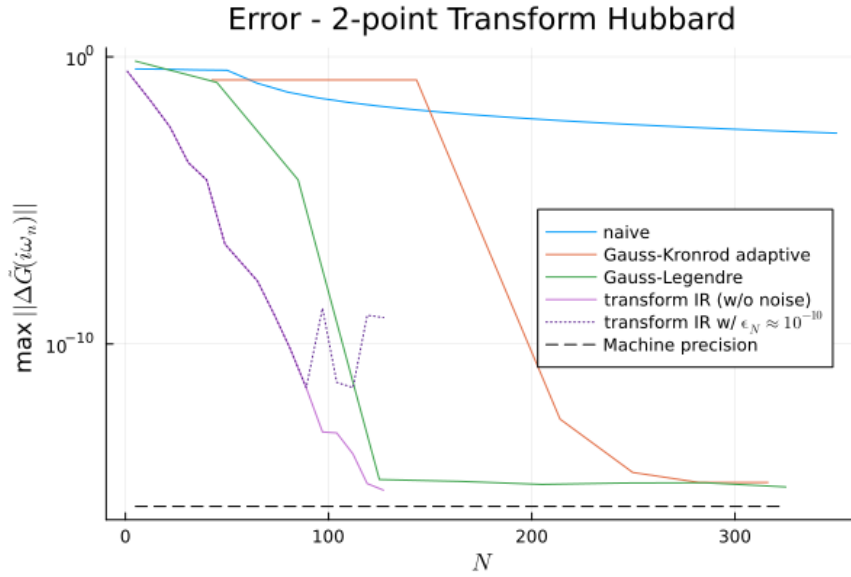


Figure 3.4: Maximum absolute error for each two-point transformation method. The maximum norm of the difference between the imaginary parts was used. For Gauss-Kronrod, the number of points for each ω was averaged. For IR, the number of basis functions represents N .

the calculated naive points did not take too long but also weren't very accurate. The other methods took a bit longer but were able to become highly accurate. Note that the IR method takes less time in the beginning, but needs more runtime to get below 10^{-10} . It therefore needs the least number of sampling points, but the longest runtime for maximum accuracy in the two-point function case. Lastly, we measured the runtime of each method with respect to the number of sampling points. Figure 3.6 shows the time in seconds for each method with respect to the number of points used.

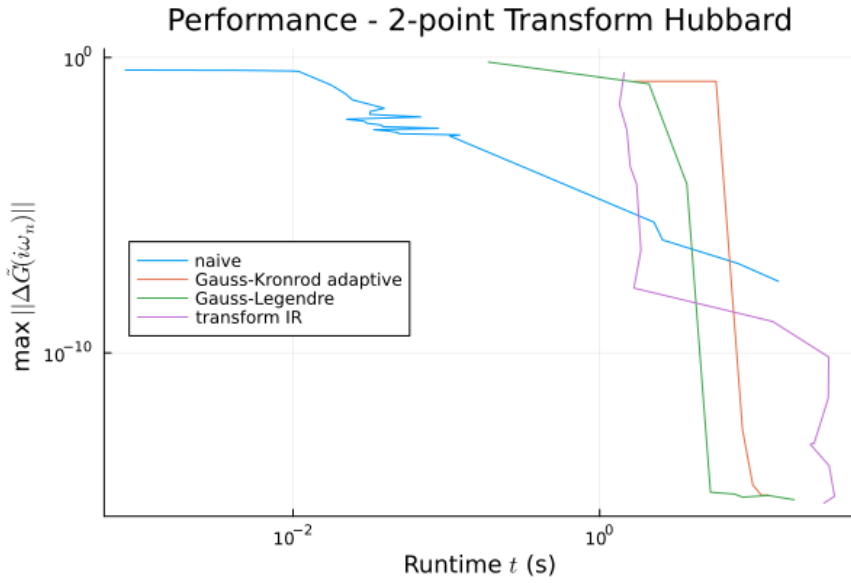


Figure 3.5: Maximum absolute error as a function of computation time t . The double log plot shows which methods are both fast as well as accurate.

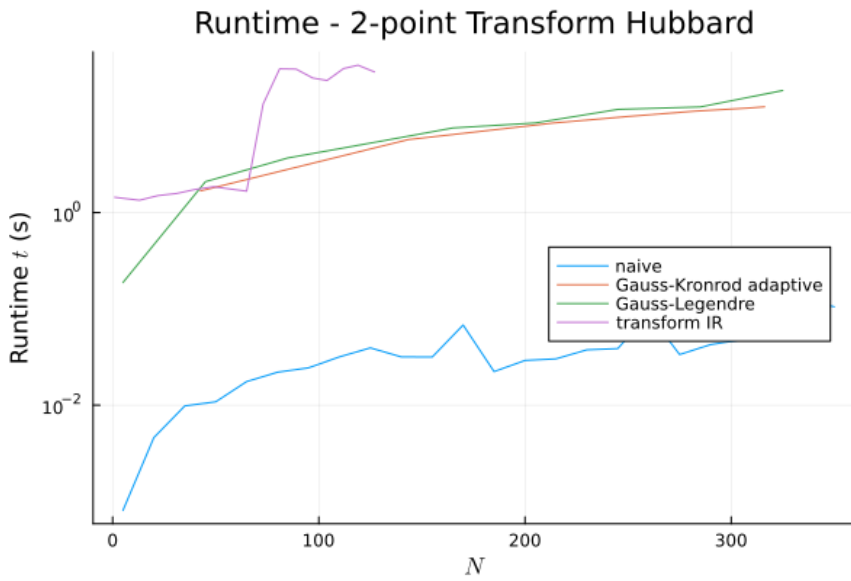


Figure 3.6: Runtime in seconds for each method with respect to the number of sampling points. The runtime for each N was measured while computing the data in Figure 3.4.

4 Three-Point Green's Function

We continue by studying the different ways of transforming three-point Green's functions to frequency space. Three-point propagator functions consist of the three operators \hat{A} , \hat{B} and \hat{C} , which are propagated through the two times τ_1 and τ_2 . We again calculate the Green's function as an example on a two-site Hubbard (see Equation (1.16)), again with the parameters $U = 3$, $t = 1$ and half-filling with $\mu = 1.5$. The temperature is again $\beta = 10$. We will use $(\hat{A}, \hat{B}, \hat{C}) = (\hat{c}_1^{\dagger}, \hat{n}_2^{\dagger}, \hat{c}_1^{\dagger})$. For these parameters, the Lehmann representation of the three-point Green's function discussed in the next section does look like in Figure 4.1.

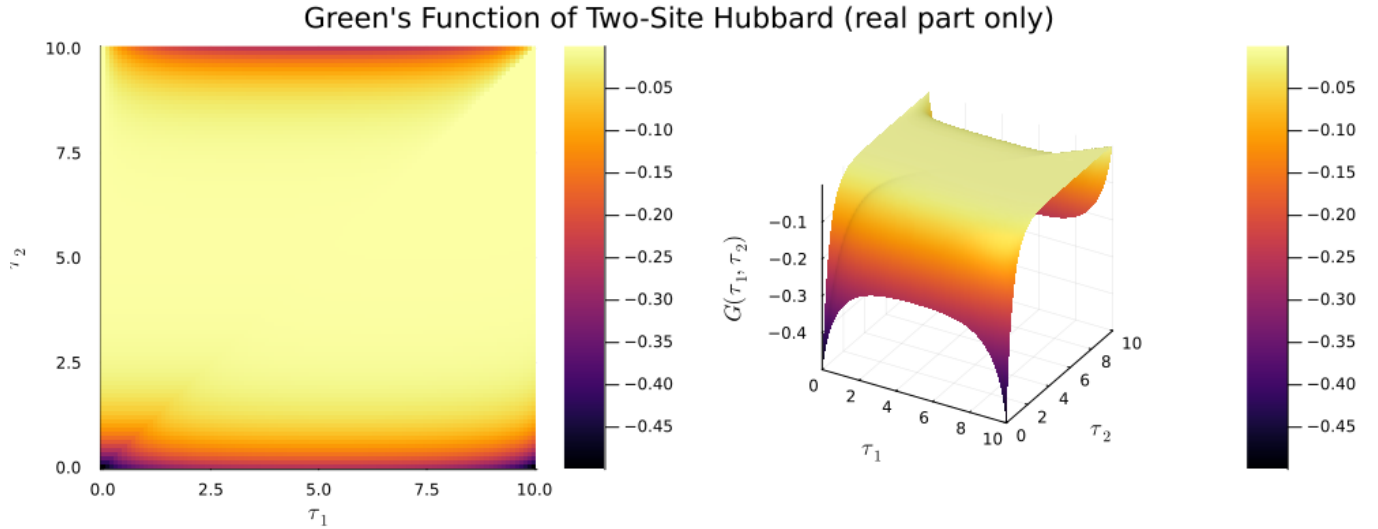


Figure 4.1: Real part of the exact Green's function (see Equation (4.6)) of the two-site Hubbard model for $U = 3$, $\beta = 10$, $t = 1$ and $\mu = 1.5$. The imaginary part is not shown here, due to being zero.

4.1 Lehmann Representation

We now derive the Lehmann representation for a three-point structure. The three-point Green's function with three operators and two times can be written as

$$G_{ABC}(\tau_1, \tau_2) = \hbar T \hat{A}(\tau_1) \hat{B}(\tau_2) \hat{C}(0) i \quad (4.1)$$

with the time ordering operator T and $0 \leq \tau_1, \tau_2 \leq \beta$. This can be written as

$$G_{ABC}(\tau_1, \tau_2) = \frac{1}{Z} \text{Tr} \left[e^{-\beta \hat{H}} \hat{A}(\tau_1) \hat{B}(\tau_2) \hat{C}(0) \right] i\theta(\tau_1 - \tau_2) + \frac{1}{Z} \text{Tr} \left[e^{-\beta \hat{H}} \hat{B}(\tau_2) \hat{A}(\tau_1) \hat{C}(0) \right] i\theta(\tau_2 - \tau_1) \quad (4.2)$$

in the Heisenberg picture. In the Schrödinger picture, this yields:

$$G_{ABC}(\tau_1, \tau_2) = \frac{1}{Z} \text{Tr} \left[e^{-\beta \hat{H}} \hat{A} e^{-\tau_1 \hat{H}} e^{\tau_2 \hat{H}} \hat{B} e^{-\tau_2 \hat{H}} \hat{C} \right] i\theta(\tau_1 - \tau_2) + \frac{1}{Z} \text{Tr} \left[e^{-\beta \hat{H}} \hat{B} e^{-\tau_2 \hat{H}} e^{\tau_1 \hat{H}} \hat{A} e^{-\tau_1 \hat{H}} \hat{C} \right] i\theta(\tau_2 - \tau_1) \quad (4.3)$$

Calculating the expectation value of Equation (4.3) and using basis completeness $\mathbf{1} = \sum_s |s\rangle \langle s|$, where $|s\rangle$ is an eigenvector of the Hamiltonian $\hat{H}|s\rangle = \epsilon_s |s\rangle$, gives

$$\begin{aligned} G_{ABC}(\tau_1, \tau_2) &= \frac{1}{Z} \text{Tr} \left[e^{-\beta \hat{H}} \hat{A} e^{-\tau_1 \hat{H}} \hat{B} e^{\tau_2 \hat{H}} \hat{C} \right] i\theta(\tau_1 - \tau_2) + \frac{1}{Z} \text{Tr} \left[e^{-\beta \hat{H}} \hat{B} e^{-\tau_2 \hat{H}} \hat{A} e^{\tau_1 \hat{H}} \hat{C} \right] i\theta(\tau_2 - \tau_1) \\ &= \frac{1}{Z} \sum_{n,m,s} \langle n | \hat{A} e^{-\tau_1 \hat{H}} | m \rangle \langle m | \hat{B} e^{\tau_2 \hat{H}} | s \rangle \langle s | \hat{C} | n \rangle i\theta(\tau_1 - \tau_2) \\ &\quad + \frac{1}{Z} \sum_{n,m,s} \langle n | \hat{B} e^{-\tau_2 \hat{H}} | m \rangle \langle m | \hat{A} e^{\tau_1 \hat{H}} | s \rangle \langle s | \hat{C} | n \rangle i\theta(\tau_2 - \tau_1) \\ &= \frac{1}{Z} \sum_{n,m,s} \langle n | \hat{A} e^{-\tau_1 \hat{H}} | m \rangle \langle m | \hat{B} e^{\tau_2 \hat{H}} | s \rangle \langle s | \hat{C} | n \rangle i\theta(\tau_1 - \tau_2) \\ &\quad + \frac{1}{Z} \sum_{n,m,s} \langle n | \hat{B} e^{-\tau_2 \hat{H}} | m \rangle \langle m | \hat{A} e^{\tau_1 \hat{H}} | s \rangle \langle s | \hat{C} | n \rangle i\theta(\tau_2 - \tau_1) \end{aligned} \quad (4.4)$$

$$\begin{aligned} &= \frac{1}{Z} \sum_n e^{-\beta \epsilon_n} \sum_{m,s} e^{-\tau_1(\epsilon_m - \epsilon_n)} e^{\tau_2(\epsilon_s - \epsilon_m)} \langle n | \hat{A} | m \rangle \langle m | \hat{B} | s \rangle \langle s | \hat{C} | n \rangle i\theta(\tau_1 - \tau_2) \\ &\quad + \frac{1}{Z} \sum_n e^{-\beta \epsilon_n} \sum_{m,s} e^{-\tau_2(\epsilon_m - \epsilon_n)} e^{\tau_1(\epsilon_s - \epsilon_m)} \langle n | \hat{B} | m \rangle \langle m | \hat{A} | s \rangle \langle s | \hat{C} | n \rangle i\theta(\tau_2 - \tau_1) \end{aligned} \quad (4.5)$$

Equation (4.5) can be written as

$$G_{ABC}(\tau_1, \tau_2) = \frac{1}{Z} \sum_n G_{ABC}^{(n)}(\tau_1, \tau_2) \quad (4.6)$$

with

$$\begin{aligned} G_{ABC}^{(n)}(\tau_1, \tau_2) &= e^{-\beta \epsilon_n} \sum_{m,s} e^{-\tau_1(\epsilon_m - \epsilon_n)} e^{\tau_2(\epsilon_s - \epsilon_m)} \langle n | \hat{A} | m \rangle \langle m | \hat{B} | s \rangle \langle s | \hat{C} | n \rangle i\theta(\tau_1 - \tau_2) \\ &\quad + e^{-\beta \epsilon_n} \sum_{m,s} e^{-\tau_2(\epsilon_m - \epsilon_n)} e^{\tau_1(\epsilon_s - \epsilon_m)} \langle n | \hat{B} | m \rangle \langle m | \hat{A} | s \rangle \langle s | \hat{C} | n \rangle i\theta(\tau_2 - \tau_1) \end{aligned} \quad (4.7)$$

4.2 Analytic (Fermionic) Transformation

Transforming the three-point Green's function into the frequency domain shows interesting behaviour, as it is the first type of correlation function which introduces a discontinuity in the middle of the to-be-integrated area (see Figure 4.2). We split

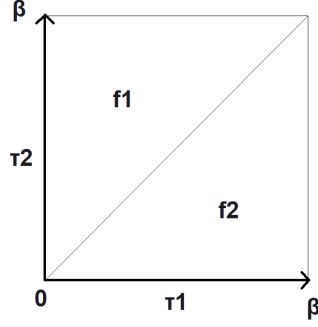


Figure 4.2: Two-dimensional integration area. Due to the theta functions in Equation (4.7), the area is split into two simplices (triangles).

the function into two terms T_1 and T_2 :

$$\begin{aligned}
 T_1 &= e^{-\beta\epsilon_n} \sum_{m,s} \underbrace{e^{-\tau_1(\epsilon_m - \epsilon_n)} e^{-\tau_2(\epsilon_s - \epsilon_m)}}_{\text{This part depends on } \tau_i} \langle m_j A_j m_i | h m_j B_j s_i | h s_j C_j n_i \rangle \\
 T_2 &= e^{-\beta\epsilon_n} \sum_{m,s} \underbrace{e^{-\tau_2(\epsilon_m - \epsilon_n)} e^{-\tau_1(\epsilon_s - \epsilon_m)}}_{\text{This part depends on } \tau_i} \langle m_j B_j m_i | h m_j A_j s_i | h s_j C_j n_i \rangle
 \end{aligned} \tag{4.8}$$

With this, we can write the n-th Fourier-transformed three-point Green's function (see Equation (4.7)) as the following:

$$\mathcal{G}_{ABC}^{(n)}(\omega_\nu, \omega_\mu) = \iint_{[0, \beta]^2 = f1 \cup f2} G_{ABC}^{(n)}(\tau_1, \tau_2) e^{i\omega_\nu \tau_1} e^{i\omega_\mu \tau_2} \tag{4.9}$$

$$= \underbrace{\left(\int_0^\beta d\tau_1 \int_0^{\tau_1} d\tau_2 T_2 \right)}_{\tau_2 \tau_1} + \underbrace{\left(\int_0^\beta d\tau_2 \int_0^{\tau_2} d\tau_1 T_1 \right)}_{\tau_1 \tau_2} e^{i\omega_\nu \tau_1} e^{i\omega_\mu \tau_2} \tag{4.10}$$

With $\tau_{\pi_1 \pi_2} = \int_0^{\tau_{\pi_1}} \int_0^{\tau_{\pi_2}} \beta g$, the two subareas f1 and f2 in Figure 4.2 can be described as f1 = τ_{12} and f2 = τ_{21} , because in f1, $\tau_1 \geq \tau_2$, while in f2, $\tau_2 \geq \tau_1$ ($\tau_{12} \cup \tau_{21} = [0, \beta]^2$). As one can see in Equation (4.9), applying the Fourier-transforming integrals to the n-th Green's function can be written as the sum of both function parts T_1, T_2 integrated in their respective domain f1, f2. By inspecting Figure 4.2, one can see that the order of integration depends whether

τ_1 or τ_2 is bigger.

For solving the integrals, only the parts of Equation (4.8) depending on τ_i are relevant. Therefore, we just have to integrate those parts. We have to solve the two integrals

$$\begin{aligned} I_1 &= \int_0^\beta d\tau_1 \int_0^{\tau_1} d\tau_2 e^{-\tau_1(\epsilon_m - \epsilon_n)} e^{-\tau_2(\epsilon_s - \epsilon_m)} e^{i\omega_\nu \tau_1} e^{i\omega_\mu \tau_2} \\ &= \int_0^\beta d\tau_1 e^{-\tau_1(\epsilon_m - \epsilon_n)} e^{i\omega_\nu \tau_1} \int_0^{\tau_1} d\tau_2 e^{-\tau_2(\epsilon_s - \epsilon_m)} e^{i\omega_\mu \tau_2} \end{aligned} \quad (4.11)$$

and

$$\begin{aligned} I_2 &= \int_0^\beta d\tau_2 \int_0^{\tau_2} d\tau_1 e^{-\tau_2(\epsilon_m - \epsilon_n)} e^{-\tau_1(\epsilon_s - \epsilon_m)} e^{i\omega_\nu \tau_1} e^{i\omega_\mu \tau_2} \\ &= \int_0^\beta d\tau_2 e^{-\tau_2(\epsilon_m - \epsilon_n)} e^{i\omega_\mu \tau_2} \int_0^{\tau_2} d\tau_1 e^{-\tau_1(\epsilon_s - \epsilon_m)} e^{i\omega_\nu \tau_1} \end{aligned} \quad (4.12)$$

We begin by solving Equation (4.11). Combining the exponential functions and evaluating the second integral gives

$$I_1 = \int_0^\beta d\tau_1 e^{-\tau_1(\epsilon_m - \epsilon_n - i\omega_\nu)} \underbrace{\int_0^{\tau_1} d\tau_2 e^{-\tau_2(\epsilon_s - \epsilon_m - i\omega_\mu)}}_{\frac{1}{(\epsilon_s - \epsilon_m - i\omega_\mu)} (e^{-\tau_1(\epsilon_s - \epsilon_m - i\omega_\mu)} - 1)} \quad (4.13)$$

$$= \frac{1}{(\epsilon_s - \epsilon_m - i\omega_\mu)} \int_0^\beta d\tau_1 \underbrace{\left(e^{-\tau_1(\epsilon_m - \epsilon_n - i\omega_\nu)} e^{-\tau_1(\epsilon_s - \epsilon_m - i\omega_\mu)} \right)}_{e^{-\tau_1(\epsilon_s - \epsilon_n - i\omega_\nu - i\omega_\mu)} - 1} \quad (4.14)$$

Before we can evaluate the last integral in Equation (4.14), we have to acknowledge that the first exponential term can indeed be one. In this case, it is just integrated to β , a special case which we will include in the final formula for the transformed Green's function. If $\epsilon_s - \epsilon_n - i\omega_\nu - i\omega_\mu \neq 0$, we integrate it to $\frac{1}{(\epsilon_s - \epsilon_n - i\omega_\nu - i\omega_\mu)} (e^{-\beta(\epsilon_s - \epsilon_n - i\omega_\nu - i\omega_\mu)} - 1)$. Therefore,

$$I_1 = \frac{1}{(\epsilon_s - \epsilon_m - i\omega_\mu)} \left(\int_0^\beta d\tau_1 e^{-\tau_1(\epsilon_s - \epsilon_n - i\omega_\nu - i\omega_\mu)} - \int_0^\beta d\tau_1 e^{-\tau_1(\epsilon_m - \epsilon_n - i\omega_\nu)} \right) \quad (4.15)$$

$$= \frac{1}{(\epsilon_s - \epsilon_m - i\omega_\mu)} (X(\epsilon_s, \epsilon_n, \omega_\nu, \omega_\mu) - \frac{1}{(\epsilon_m - \epsilon_n - i\omega_\nu)} (e^{-\beta(\epsilon_m - \epsilon_n - i\omega_\nu)} - 1)) \quad (4.16)$$

with

$$X = \begin{cases} \beta, & \text{if } \epsilon_s = \epsilon_n = i\omega_\nu = i\omega_\mu = 0 \\ \frac{1}{(\epsilon_s - \epsilon_n - i\omega_\nu - i\omega_\mu)} (e^{-\beta(\epsilon_s - \epsilon_n - i\omega_\nu - i\omega_\mu)} - 1) & \text{otherwise} \end{cases} \quad (4.17)$$

Doing the same procedure for integral I_2 gives us

$$I_2 = \frac{1}{(\epsilon_s - \epsilon_m - i\omega_\nu)} (X(\epsilon_s, \epsilon_n, \omega_\nu, \omega_\mu) - \frac{1}{(\epsilon_m - \epsilon_n - i\omega_\mu)} (e^{-\beta(\epsilon_m - \epsilon_n - i\omega_\nu)} - 1)) \quad (4.18)$$

with the same conditional X . Note that when comparing I_1 and I_2 , we find that

$$I_1(\omega_\nu, \omega_\mu) = I_2(\omega_\mu, \omega_\nu) \quad (4.19)$$

Therefore, we write the Fourier-transformed three-point Green's function as

$$\mathcal{G}_{ABC}(\omega_\nu, \omega_\mu) = \frac{1}{Z} \sum_n \mathcal{G}_{ABC}^{(n)}(\omega_\nu, \omega_\mu) \quad (4.20)$$

with

$$\begin{aligned} \mathcal{G}_{ABC}^{(n)}(\omega_\nu, \omega_\mu) = & e^{-\beta\epsilon_n} \sum_{m,s} I_1 \langle m | A | j \rangle \langle m | B | s \rangle \langle j | C | n \rangle \\ & e^{-\beta\epsilon_n} \sum_{m,s} I_2 \langle m | B | j \rangle \langle m | A | s \rangle \langle j | C | n \rangle \end{aligned} \quad (4.21)$$

With I_1 defined in Equation (4.16) and I_2 defined in Equation (4.18), which both use the definition for $X(\epsilon_s, \epsilon_n, \omega_\nu, \omega_\mu)$ (Equation (4.17)). When operators \hat{A} and \hat{B} are both fermionic, the second term in Equation (4.21) becomes negative due to Wick parity. If that is the case, also $e^{\beta i\omega_\nu} = -1$ and $e^{\beta i\omega_\mu} = -1$, which we will use in our computations. Figure 4.3 shows the plotted analytically transformed propagator of our two-site Hubbard Hamiltonian. Note that in (4.20) and (4.21), we have a triple sum over all eigenstates, which makes the computation very slow for larger systems.

4.3 Naive Transformation

Again, we try to simply slice up the integrals into sums as the simplest approach for calculation.

$$\mathcal{G}(\omega_\nu, \omega_\mu) = \sum_l \sum_j e^{i\omega_\nu \tau_l} e^{i\omega_\mu \tau_j} \mathcal{G}(\tau_l, \tau_j) \quad \tau_j \geq \tau_l \quad (4.22)$$

Remember that now, we have a grid in two dimensions. Therefore, the computational expense increases quadratically with the number of tau points (and quadratically with the number of Matsubara frequencies).

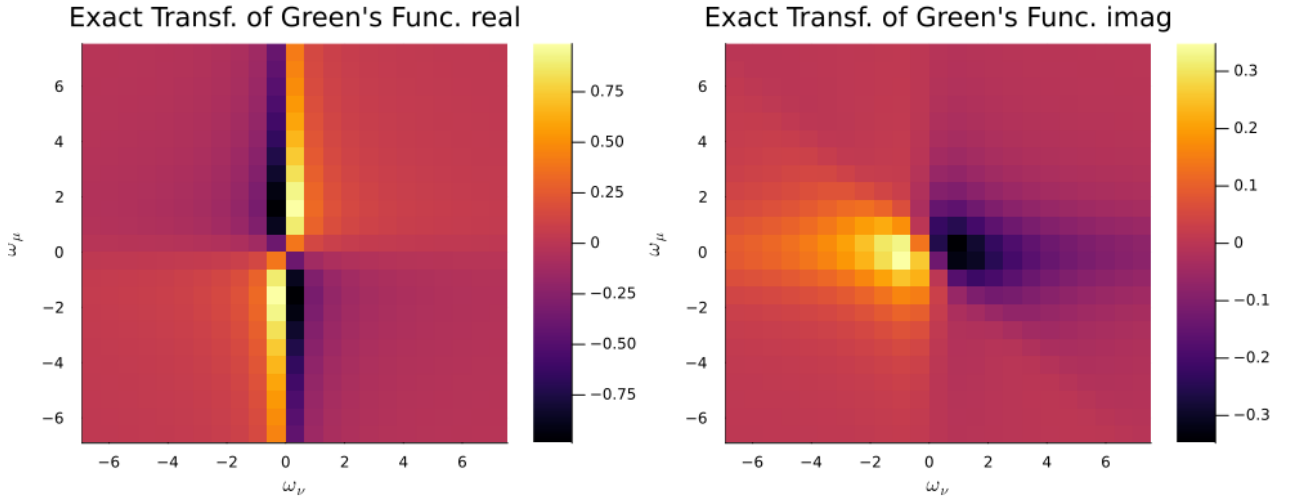


Figure 4.3: Real and imaginary part of the analytically transformed exact Green's function (see Equation (4.20)) of the two-site Hubbard model with the operators and parameters as in Figure 4.1. Note that the Matsubara frequencies are discrete. The real part is not zero.

4.4 Gauss Quadrature

Analogous to the two-point function, the next step is to introduce weights.

$$G(\omega_\nu, \omega_\mu) = \sum_l \sum_j w_l w_j e^{i\omega_\nu \tau_l} e^{i\omega_\mu \tau_j} G(\tau_l, \tau_j) \quad \tau_j \quad \tau_l \quad (4.23)$$

However, in the three-point function, there is a catch. When calculating the Lehmann representation, we already have seen in Figure 4.2, the integration area splits into two triangles. In the diagonal between them, the function is non-continuous, which results in most integration algorithms having problems when computing the β β quad. The triangles are geometrically also called simplices. Both leg lengths are equal and the angle between them is 90 degrees. Special simplex integration algorithms exist in order to integrate simplices, therefore, we just split the integration area into these two simplices and afterwards add the results together. The used simplex integration algorithms are SimplexQuad.jl and GrundmannMoeller.jl. The Grundmann-Möller method overshoots at the edges, therefore, a cutoff function was developed (explained in Algorithm 1), which solved the problem partially (see Figure 4.4). Again, we refer to the literature (resp. to the original code in case of SimplexQuad) [7] [8].

Algorithm 1 Cutoff algorithm used to get meaningful results from the Grundmann-Möller transformation method for fermionic Green's functions. It searches for the minimum and maximum value in the neighborhood of $G(0,0)$, and removes all other values which are bigger/smaller than these. The algorithm was applied to the real and imaginary parts separately.

```

1:  $x_{\max}$     $\max(G(\omega_a, \omega_b))$  for  $a, b \in [-2, -1, 0, 1, 2]$ 
2:  $x_{\min}$     $\min(G(\omega_a, \omega_b))$  for  $a, b \in [-2, -1, 0, 1, 2]$ 
3: for all  $g \in G(\omega_\nu, \omega_\mu)$  do
4:   if  $g > 0 \ \&\& \ g > x_{\max}$  then
5:      $g = 0$ 
6:   else if  $g < 0 \ \&\& \ g < x_{\min}$  then
7:      $g = 0$ 
8:   end if
9: end for

```

4.5 Analysis

The same methods as in Chapter 3 are used for error analysis. Due to the results being two-dimensional, the difference plot has to be done either with a projection (Figure 4.8) or by using different plotting methods (Figure 4.5, 4.6 and 4.7). Figure 4.9 shows the log plot of the error with respect to the corresponding sampling points. Again, the maximum norm (see Equation (3.25)) was used. Grundmann-Möller becomes even less accurate than naive, but for both, the maximum norm does not really decrease with increasing function evaluation points. SimplexQuad however is working fine. The absolute error which can also be interpreted as the accuracy as a function of time is again shown in Figure 4.10 for the three-point fermionic case. Note that the naive as well as the Grundmann-Möller methods have a hard time trying to become more accurate with increasing runtime. An explanation is probably the bumps both methods have around $\omega_\nu = 0$, which can be seen in Figure 4.5 and 4.7. Due to the maximum norm being used, the respective bumps only decrease slowly. However, the SimplexQuad error quickly decreases after 10s. The runtime was also computed for each method with respect to the number of sampling points. Figure 4.11 shows the time in seconds for each method with respect to the number of points used.

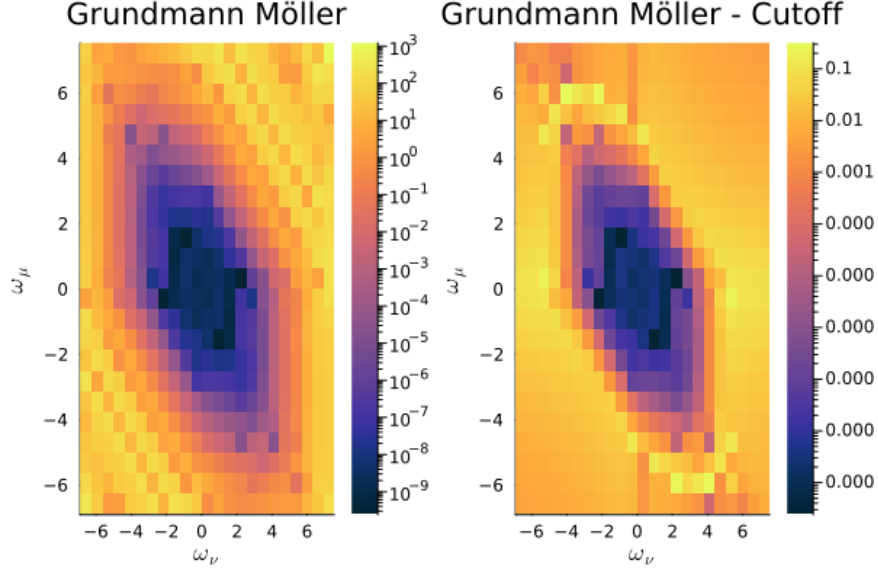


Figure 4.4: *Left*: $j\text{Im}[\mathcal{G}_{\text{Grundmann-Möller}}(\omega_\nu, \omega_\mu)] - \text{Im}[\mathcal{G}_{\text{Lehmann}}(\omega_\nu, \omega_\mu)]j$ before the cutoff. As can be clearly seen from the log color scale, the diagonal edges overshoot to infinity. *Right*: After the cutoff (Algorithm 1). Still, the bump at $\omega_\nu = 0$ could not be removed. Due to a plot error, the log scale is not shown correctly, and the negative power of tens lack additional 0s and 1s.

4.6 Nonlinear (Bosonic) Response with exact Diagonalization

As a different example of the application of numerical Fourier Transform we consider nonlinear density response in the Anderson impurity model (AIM) [11]. We now consider bosonic operators in the correlation function and a new Hamiltonian.

$$H = \mu \sum_{\sigma} \hat{c}_{1\sigma}^{\dagger} \hat{c}_{1\sigma} + U \hat{n}_1 \hat{n}_{1\neq} + \sum_{i>1, \sigma} V_{1i} (\hat{c}_{1\sigma}^{\dagger} \hat{c}_{i\sigma} + \hat{c}_{i\sigma}^{\dagger} \hat{c}_{1\sigma}) - \sum_{i>1, \sigma} \epsilon_i \hat{c}_{i\sigma}^{\dagger} \hat{c}_{i\sigma} \quad (4.24)$$

Equation (4.24) shows the general Hamiltonian of the Anderson impurity model, with one impurity ($i = 1$) and $n - 1$ bath sites. The impurity site is described as the one-site model in Equation (1.17). The bath sites have a certain bath dispersion ϵ_i and are connected to the impurity via the hybridization V_{1i} [2]. For the practical computations, we will use a three-site bath ($n = 4$) in order to keep the computational complexity for the Lehmann representation manageable. We will use the parameters $U = 2$, $\mu = 1.33168$, and for the bath sites $\epsilon_i = [1.59210263 \cdot 10^0, 1.59477752 \cdot 10^{-3}, 1.82320307 \cdot 10^0]$ and $V_{1i}^2 =$

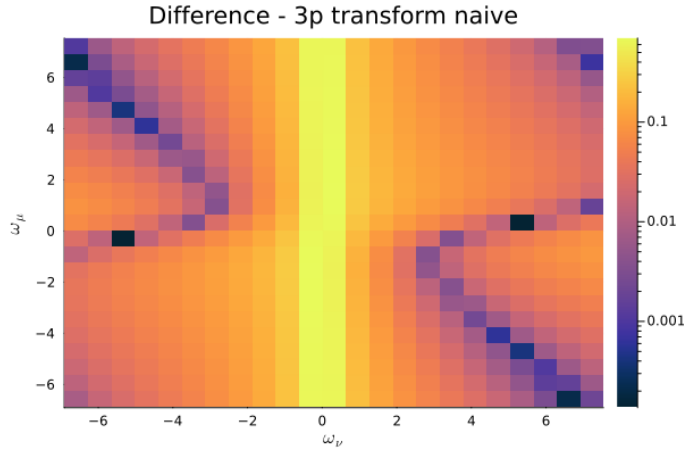


Figure 4.5: Absolute error in the imaginary part of the naive transform with $N = 1250$ total sampling points, with fermionic operators on the Hubbard model. Note the big bump at $\omega_\nu = 0$. Manual tests have shown that this bump only vanishes very slowly, with exponentially more used sampling points.

[$1.59210263 \cdot 10^0$, $1.59477752 \cdot 10^{-3}$, $1.82320307 \cdot 10^0$]. These values come from the fact that via a method called dynamic mean field theory (DMFT), a Hubbard model can be represented by a self-consistent solution for the AIM, which was done in this case. Further, we will use the three different operator combinations: $G_{R1} = \hbar T \hat{A} \hat{B} \hat{C} i = \hbar T \hat{n}_1 \hat{n}_1 \hat{n}_1 i$, $G_{R4} = \hbar T \hat{n}_1 \hat{n}_1 \hat{n}_1 \hat{i}$ and $G_{R7} = \hbar T \hat{n}_1 \hat{c}_1^\dagger \hat{c}_1 \hat{c}_1^\dagger \hat{c}_1 i$. Figure 4.12 shows the three correlation functions on the fitted AIM model in frequency space. Note that the imaginary parts of $R1$ and $R4$ are noisy but practically zero, while the imaginary part of $R7$ does not vanish.

We will start again by deriving the Lehmann representation, which in case of bosons is not entirely the same solution as Equation (4.7). The background is the following; due to the bosonic Matsubara frequencies being even instead of uneven (in the fermionic case), not only do the frequencies become $\omega_x = \frac{2x}{\beta} \pi$ and with that $e^{i\beta\omega_x} = 1$, but also other exponential terms can become constant ($e^{-\tau\omega_x} = 1$) due to the even case of $x = 0$. Therefore, the integration has to be done with even more caution than in Chapter 4.2 in order to capture the additional edge cases where the integration could yield a different result.

For simplicity, we again start with the initial integral I_1 in Equation (4.13):

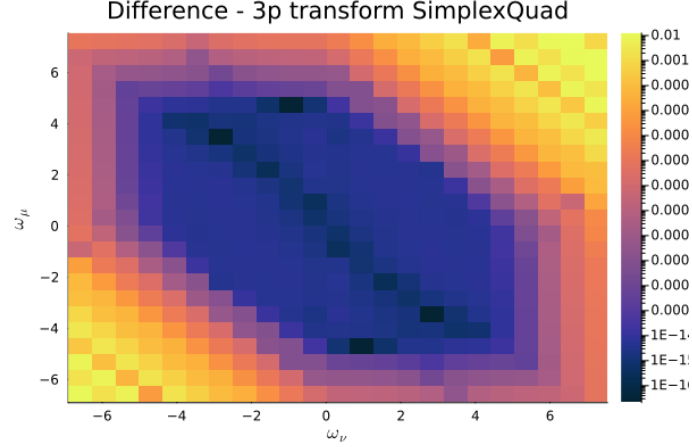


Figure 4.6: Absolute error in the imaginary part of the SimplexQuad method and $N = 1250$ sampling points, with fermionic operators on the Hubbard model. Like in Figure 4.4, the proceeding negative powers of ten in the color log scale are not shown correctly after 0.001.

$$I_1 = \int_0^\beta d\tau_1 e^{-\tau_1(\epsilon_m - \epsilon_n + i\omega_\nu)} \underbrace{\int_0^{\tau_1} d\tau_2 e^{-\tau_2(\epsilon_s - \epsilon_m + i\omega_\mu)}}_{\text{We will evaluate this integral.}} \quad (4.25)$$

Now with the knowledge that μ and therefore ω_μ can be zero, we integrate the mentioned integral as the following:

$$\text{Integral} = \begin{cases} \tau_1, & \text{if } \epsilon_s - \epsilon_m + i\omega_\mu = 0 \\ \frac{1}{(\epsilon_s - \epsilon_m + i\omega_\mu)} (e^{-\tau_1(\epsilon_s - \epsilon_m + i\omega_\mu)} - 1) & \text{otherwise} \end{cases} \quad (4.26)$$

To improve further readability, we from now on use delta distributions instead of case distinctions:

$$I_1 = \int_0^\beta d\tau_1 e^{-\tau_1(\epsilon_m - \epsilon_n + i\omega_\nu)} (\tau_1 \delta(\epsilon_s - \epsilon_m + i\omega_\mu) + \frac{1}{(\epsilon_s - \epsilon_m + i\omega_\mu)} (e^{-\tau_1(\epsilon_s - \epsilon_m + i\omega_\mu)} - 1)) \quad (4.27)$$

Distributing the first exponential factor to the latter terms yields for I_1 in total the sum of three integrals which have to be checked once more for edge cases.

$$\begin{aligned} &= \int_0^\beta d\tau_1 \underbrace{(e^{-\tau_1(\epsilon_m - \epsilon_n + i\omega_\nu)} \tau_1 \delta(\epsilon_s - \epsilon_m + i\omega_\mu))}_{\text{Check!}} \\ &\quad + \frac{1}{(\epsilon_s - \epsilon_m + i\omega_\mu)} \underbrace{e^{-\tau_1(\epsilon_m - \epsilon_n + i\omega_\nu)}}_{\text{Check!}} \end{aligned}$$

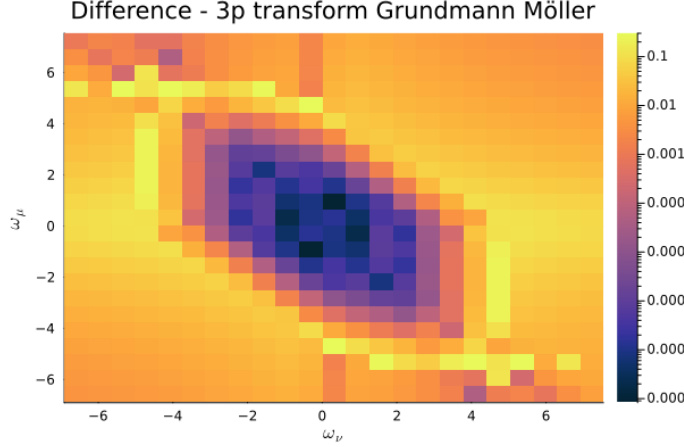


Figure 4.7: Absolute error in the imaginary part of the Grundmann-Möller method with $N = 1120$ sampling points, with fermionic operators on the Hubbard model. Note the smaller bump at $\omega_\nu = 0$ and the smaller blue convergence area. For the graphical error of the color scale on the right, see the caption of Figure 4.6.

$$\frac{1}{(\epsilon_s \quad \epsilon_m \quad i\omega_\mu)} \delta(\epsilon_s \quad \epsilon_m \quad i\omega_\mu) e^{\underbrace{\tau_1(\epsilon_s \quad \epsilon_n \quad i\omega_\nu \quad i\omega_\mu)}_{\text{Check!}}} \quad (4.28)$$

The first term requires partial integration. Evaluating (4.28) thoroughly gets us to

$$\begin{aligned} I_1 &= \delta(\epsilon_s \quad \epsilon_m \quad i\omega_\mu) \delta(\epsilon_m \quad \epsilon_n \quad i\omega_\nu) \frac{\beta^2}{2} \\ &+ \delta(\epsilon_s \quad \epsilon_m \quad i\omega_\mu) [1 \quad \delta(\epsilon_m \quad \epsilon_n \quad i\omega_\nu)] f \frac{(e^{\beta(\epsilon_m \quad \epsilon_n \quad i\omega_\nu)} \quad 1)}{(\epsilon_m \quad \epsilon_n \quad i\omega_\nu) \beta} \frac{(e^{\beta(\epsilon_m \quad \epsilon_n \quad i\omega_\nu)} \quad 1)}{(\epsilon_m \quad \epsilon_n \quad i\omega_\nu)^2} g \\ &+ \frac{1}{(\epsilon_s \quad \epsilon_m \quad i\omega_\mu)} f \delta(\epsilon_m \quad \epsilon_n \quad i\omega_\nu) \beta [1 \quad \delta(\epsilon_m \quad \epsilon_n \quad i\omega_\nu)] \frac{(e^{\beta(\epsilon_m \quad \epsilon_n \quad i\omega_\nu)} \quad 1)}{(\epsilon_m \quad \epsilon_n \quad i\omega_\nu)} \\ &\delta(\epsilon_s \quad \epsilon_n \quad i\omega_\nu \quad i\omega_\mu) \beta + [1 \quad \delta(\epsilon_s \quad \epsilon_n \quad i\omega_\nu \quad i\omega_\mu)] \frac{(e^{\beta(\epsilon_s \quad \epsilon_n \quad i\omega_\nu \quad i\omega_\mu)} \quad 1)}{(\epsilon_s \quad \epsilon_n \quad i\omega_\nu \quad i\omega_\mu)} g \quad (4.29) \end{aligned}$$

Again, it holds that $I_1(\omega_\nu, \omega_\mu) = I_2(\omega_\mu, \omega_\nu)$, see Equation (4.2). Therefore, we can again write

$$\mathcal{G}_{ABC}(\omega_\nu, \omega_\mu) = \frac{1}{Z} \sum_n \mathcal{G}_{ABC}^{(n)}(\omega_\nu, \omega_\mu) \quad (4.30)$$

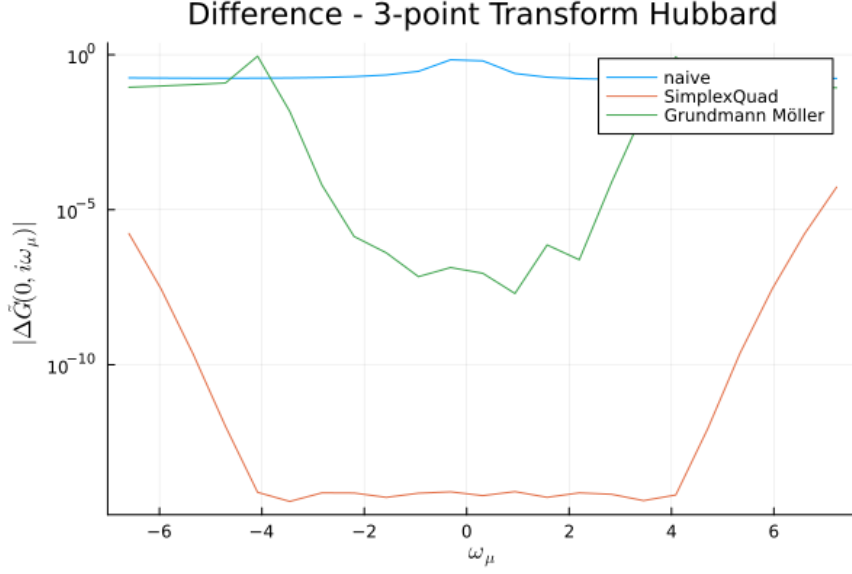


Figure 4.8: Absolute difference of the methods to the exact Lehmann transformation. The number of τ sampling points used in this case is comparable with $N_{naive} = 1250$, $N_{SimplexQuad} = 1250$, and $N_{GrundmannMoeller} = 1120$.

for the Bosonic Fourier transformed Green's function, with

$$\begin{aligned}
 G_{ABC}^{(n)}(\omega_\nu, \omega_\mu) = & e^{\beta\epsilon_n} \sum_{m,s} I_1 h m j A j m i h m j B j s i h s j C j n i \\
 & + e^{\beta\epsilon_n} \sum_{m,s} I_2 h m j B j m i h m j A j s i h s j C j n i
 \end{aligned} \tag{4.31}$$

and the I_i of this chapter. Due to the assumption of bosonic operators, we get the plus between the sums in the (n)-th Green's function. To further reduce Equation (4.29), the property $e^{i\beta\omega_x} = 1$ can be used.

4.7 Analysis (Nonlinear Response)

For the bosonic correlation function transformation, the calculation of accuracy and runtime on the presented AIM was done with the $R7$ operator combination (see Figure 4.12). For the absolute error measure, again, the maximum norm was used. Further, due to the high computational expense of the analytical solution, only three Matsubara frequency combinations were used. Figure 4.13 shows the difference between the naive, SimplexQuad and Grundmann-Möller methods

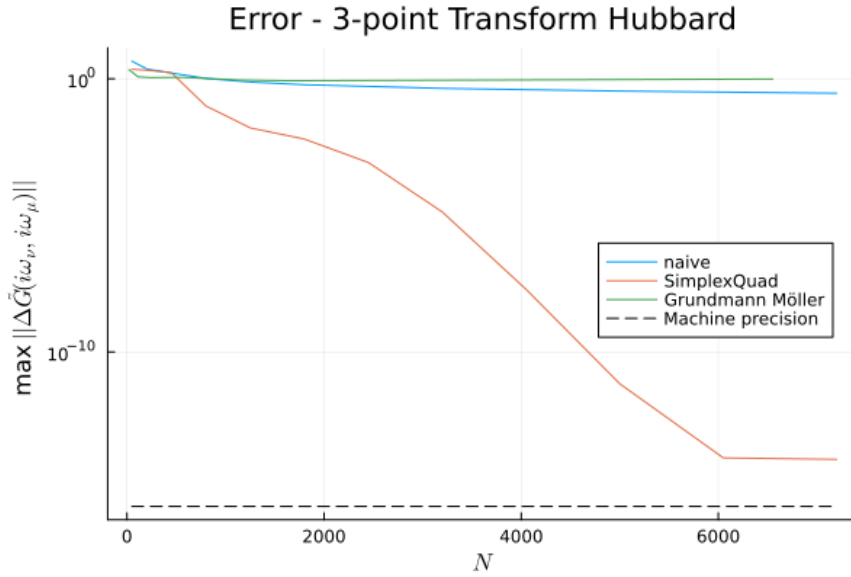


Figure 4.9: Maximum absolute error for each three-point transformation method for the fermionic three-point function on the Hubbard model.

to the Lehmann method. Note that ω_ν is varied to get three distinct values of G_{ABC} . Figure 4.14 again shows the maximum absolute error. We can see that the Grundmann-Möller method becomes more accurate than the naive transformation. Both methods seem to be more efficient than in the fermionic case, mostly probably due to the lack of bumps. Also, the cutoff did not have to be used with Grundmann-Möller. Still, the SimplexQuad integration wins, with this decreasing sub 10^{-10} . Figure 4.15 again shows the performance via the error as a function of runtime. It can be seen that the SimplexQuad integration becomes quickly extremely accurate, due to being in the left bottom corner of the plot. The other methods are slower. While the naive transform gets a little bit more accurate per runtime, the Grundmann-Möller approach becomes moderately more accurate. It is clear that these two methods are more effective in this bosonic case than in the fermionic one, compare with Figure 4.10. Regarding runtime, Figure 4.16 shows that naive and Grundmann Möller are also quite comparable in terms of runtime, while the SimplexQuad method is two powers of ten faster than the other two. We see that SimplexQuad also is fast as well as accurate with more sites as well as with bosonic operators.

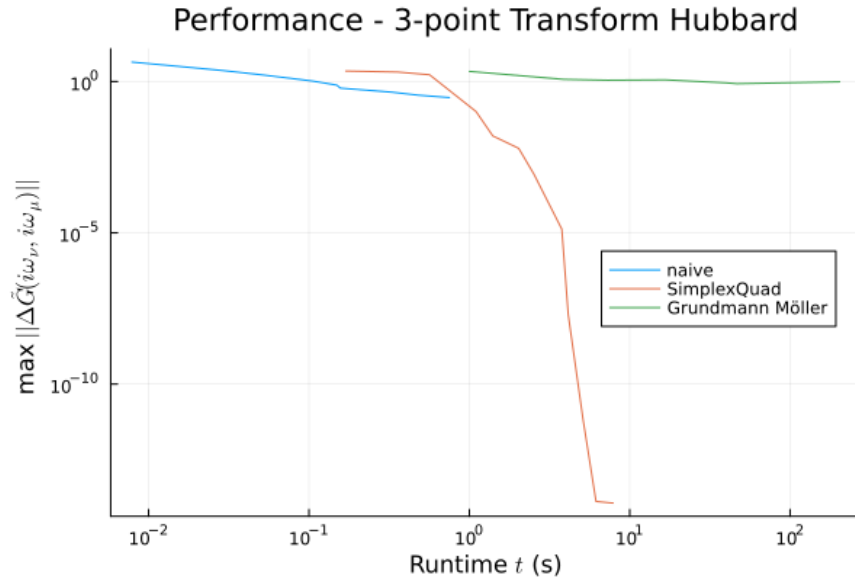


Figure 4.10: Maximum absolute error as a function of computation time t for the fermionic three-point function on the Hubbard model.

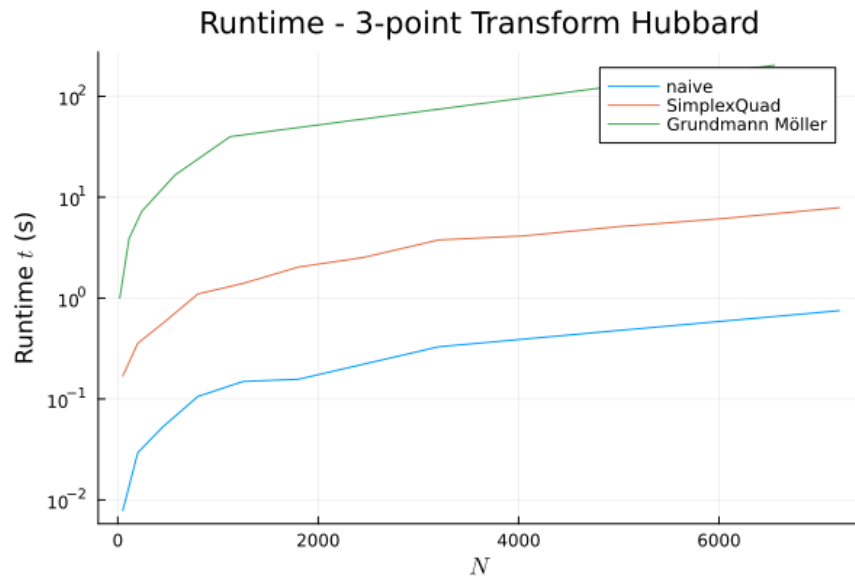


Figure 4.11: Runtime in seconds for each method with respect to the number of sampling points. The runtime for each N was measured while computing the data in Figure 4.9.

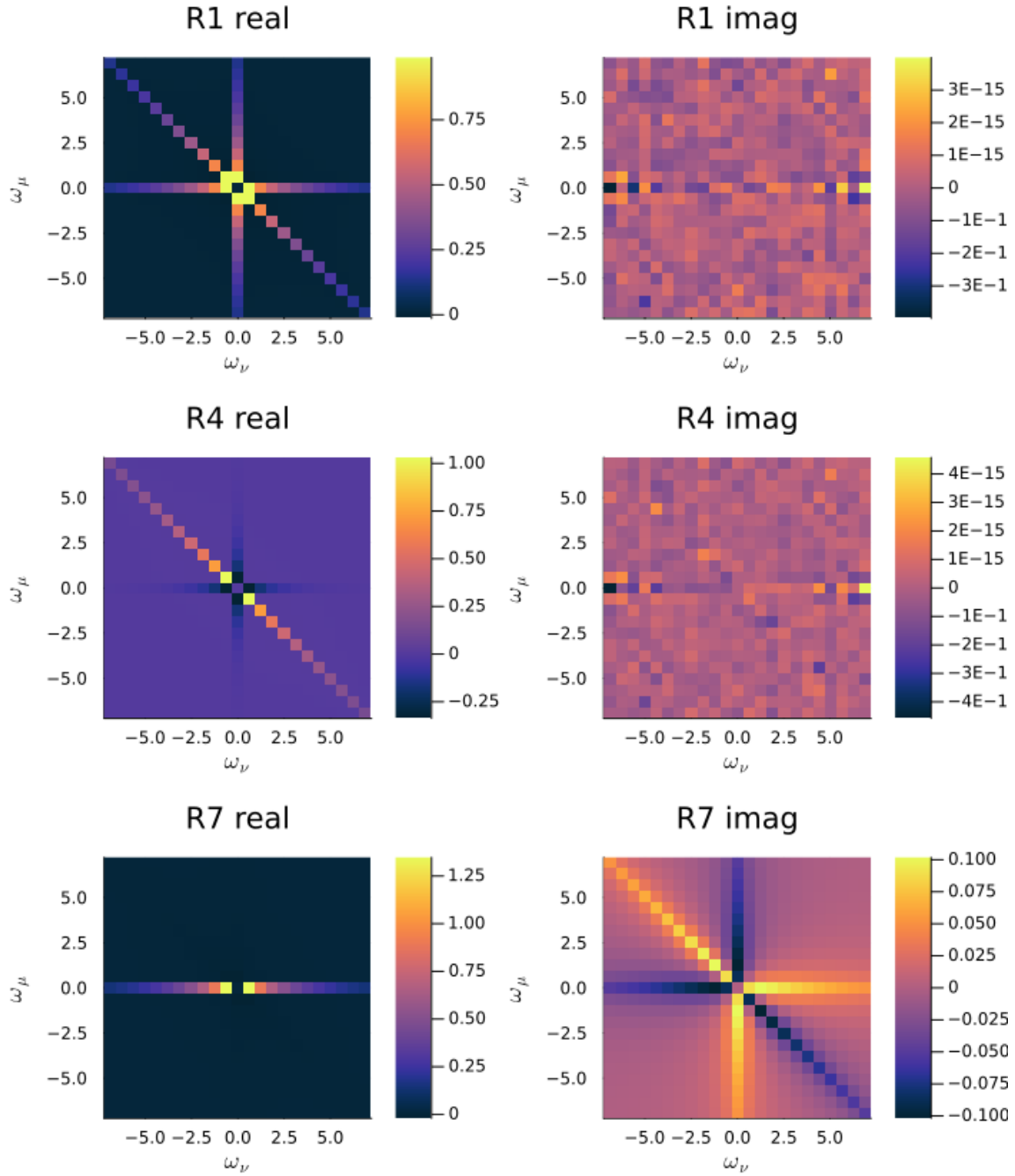


Figure 4.12: Real and imaginary parts of the three operator combinations $R1$, $R4$ and $R7$. Maxima in the middle ($G_{ABC}(0,0)$) were set to 0 to show the structure around better. $\text{real}(G_{fR1,R4,R7g}(0,0))$ (21.269, 16.783, 2.243), $\text{imag}(G_{fR1,R4,R7g}(0,0))$ (0, 0, 0.102). They were calculated with 5000 points and the SimplexQuad method, due to the site count where the Lehmann version was computationally too expensive. Note that $R1 = (\hat{A}, \hat{B}, \hat{C}) = (\hat{n}_1'', \hat{n}_1'', \hat{n}_1'')$, $R4 = (\hat{n}_1'', \hat{n}_1'', \hat{n}_1\#)$ and $R7 = (\hat{n}_1'', \hat{c}_1^y \hat{c}_{1\#}, \hat{c}_{1\#}^y \hat{c}_1'')$, $\hat{n} = \hat{c}^y \hat{c}$.

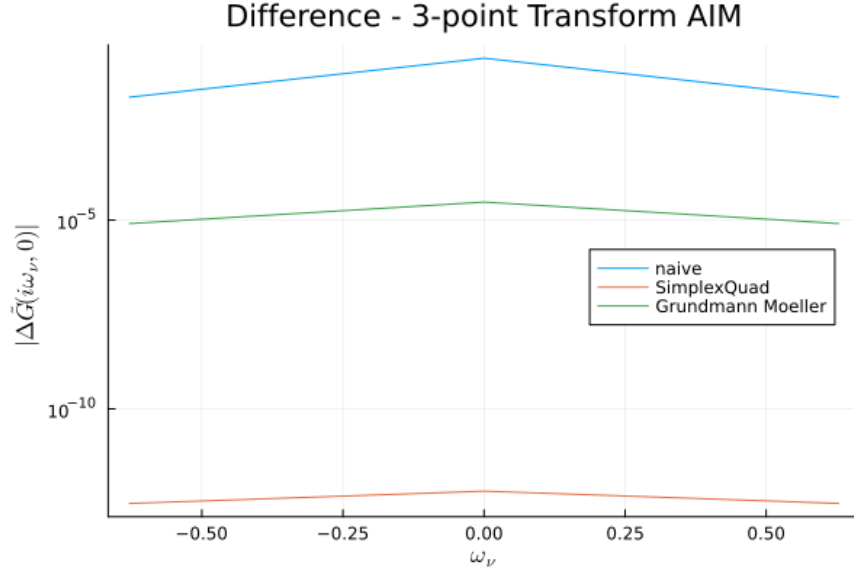


Figure 4.13: Absolute difference of the methods. The Hamiltonian used is the described fitted AIM model. The number of τ sampling points used in this case is comparable with $N_{naive} = 1250$, $N_{SimplexQuad} = 968$, $N_{GrundmannMoeller} = 1120$.

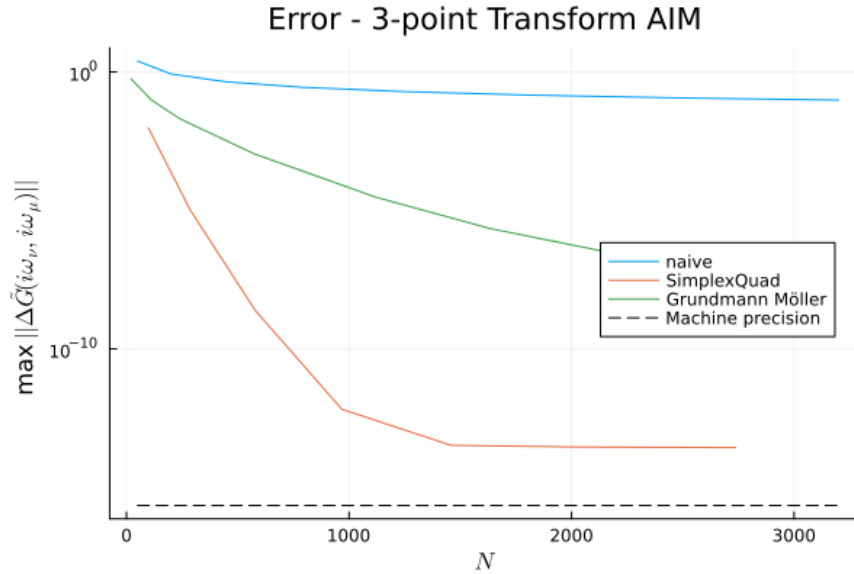


Figure 4.14: Maximum absolute error of the three methods to the Lehmann representation. The fitted AIM model was used. Because the two simplices were calculated separately, N is two times the point count used for one simplex.

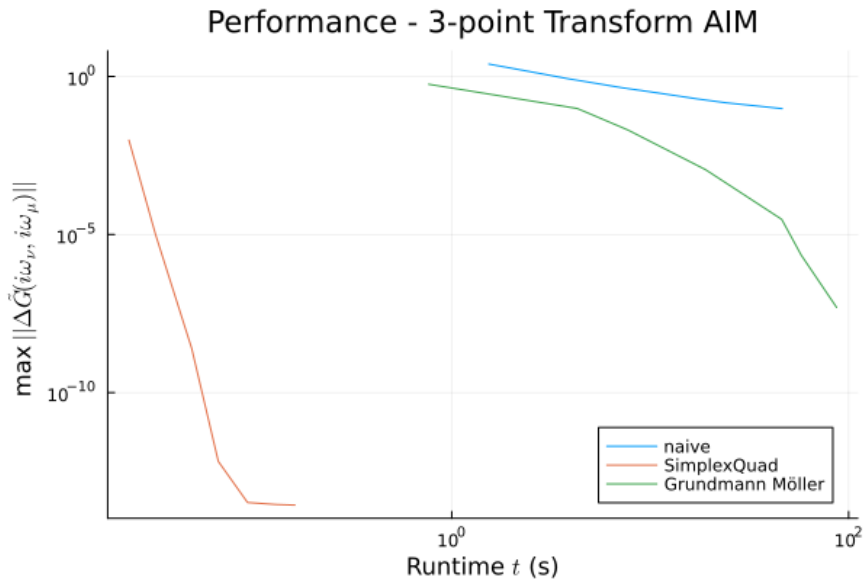


Figure 4.15: Maximum absolute error as a function of computation time t for the bosonic three-point function on the fitted Anderson impurity model.

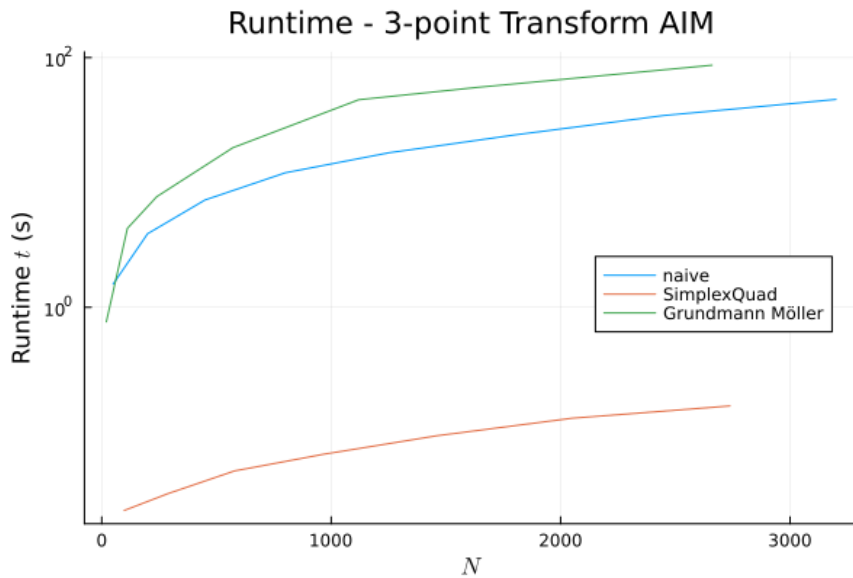


Figure 4.16: Runtime in seconds on the fitted AIM model. The SimplexQuad code was easy to vectorize, therefore, also the Green's function in imaginary time space could be calculated in one go. This could explain some of the speed benefits of this method.

5 Discussion

In this work, we looked at analytical and numerical ways of transforming Green's functions from the imaginary time domain to the frequency domain. Generally, we learn that the naive transformation is the simplest method, but also the one with the biggest maximum error, especially with time ordering. With two-point functions, multiple assessed methods are feasible and either perform best measured by the number of function evaluation points or runtime. With three-point functions, although not so many methods were studied, we find one particular method to be clearly well-performing.

In the two-point function case, the Gauss methods are much more precise around $\omega_n = 0$, and with bigger/smaller frequencies, the Nyquist theorem seems to limit the precision a lot. Gauss-Legendre has practically the same runtime as Gauss-Kronrod, and still converges a little bit faster to maximum accuracy with respect to runtime than Gauss-Kronrod. This might be due to the Gauss-Legendre order increasing with the sampling points, while the Gauss-Kronrod order is user-set and constant. Intermediate representations, while taking the most time to set up and compute, reduce the maximum error with the least number of points/basis functions due to its compressing property. Problems with this method may only appear when assumptions about the reconstructed function do not hold, as initially in our case.

The three-point functions have shown the SimplexQuad method clearly as the best method, due to it being fast as well as precise. The Grundmann Möller Simplex integration oscillates too much on the diagonal edges of the area to infinite values for the fermionic case, and even a cutoff of these extreme values did not fix the problem completely but made the method a bit better than the naive transform. In its defense, the naive integration over the whole area was also having difficulties with the discontinuity. In the bosonic case, both methods performed better. It would be really interesting to see the intermediate representation of three-point functions at this point. Also, it would be promising to apply explicit integration algorithms for discontinuous functions to the problem, which would be ideal future work tasks [24] [5] [18] [25].

Especially the nonlinear bosonic response chapter showed, due to the relative "high" number of sites used in the model, how fast the limits of practical analytic computations are reached and how relevant other transformation methods are. Still, to benchmark and understand the problem, Lehmann representation

computations were quite essential, although due to its low computation time and fast convergence, the SimplexQuad method also proved to be a very reliable tool. The transformation methods developed for fermions were also feasible for bosons and more complex models.

Bibliography

- [1] Daniel Arovav et al. “The hubbard model”. In: *Annual review of condensed matter physics* 13 (2022), pp. 239–274.
- [2] Louis-Francois Arsenault et al. “Machine learning for Many-Body Physics: the case of the Anderson impurity model”. In: *Physical Review B* 90.15, 155136 (2014).
- [3] Ignace Bogaert. “Iteration-Free Computation of Gauss–Legendre Quadrature Nodes and Weights”. In: *SIAM Journal on Scientific Computing* 36.3 (2014), A1008–A1026.
- [4] Willem Hendrik Dickhoff and Dimitri Van Neck. *Many-body theory exposed! Propagator description of quantum mechanics in many-body systems*. Singapore, World Scientific Publishing Company, 2008.
- [5] Guo-Xin Fan and Qing Huo Liu. “Fast Fourier transform for discontinuous functions”. In: *IEEE Transactions on Antennas and Propagation* 52.2 (2004), pp. 461–465.
- [6] *FastGaussQuadrature.jl*. <https://github.com/JuliaApproximation/FastGaussQuadrature.jl>. 2023.
- [7] Greg von Winckel. *n-dimensional simplex quadrature*. <https://www.mathworks.com/matlabcentral/fileexchange/9435-n-dimensional-simplex-quadrature>. 2023.
- [8] Axel Grundmann and Hans-Michael Möller. “Invariant integration formulas for the n-simplex by combinatorial methods”. In: *SIAM Journal on Numerical Analysis* 15.2 (1978), pp. 282–290.
- [9] *GrundmannMoeller.jl*. <https://github.com/eschnett/GrundmannMoeller.jl>. 2023.
- [10] *Julia*. <https://github.com/JuliaLang/Julia.jl>. 2023.
- [11] Patrick Kappl et al. “Nonlinear responses and three-particle correlators in correlated electron systems exemplified by the Anderson impurity model”. In: *Physical Review B* 107.20, 205108 (2023).
- [12] Dirk Laurie. “Calculation of Gauss-Kronrod quadrature rules”. In: *Mathematics of Computation* 66.219 (1997), pp. 1133–1145.

- [13] Richard Mattuck. *A guide to Feynman diagrams in the many-body problem*. New York, Courier Corporation, 1992.
- [14] Wolfgang von Niessen, Jochen Schirmer, and Lorenz Cederbaum. In: *Computer Physics Reports* 1.2 (1984), pp. 57–125.
- [15] *QuadGK.jl*. <https://github.com/JuliaMath/QuadGK.jl>. 2023.
- [16] Stefan Rohshap. *Analytical calculation of two-particle vertices for multiorbital Hubbard atom*. Master’s Thesis, Technische Universität Wien. 2023.
- [17] Stefan Rohshap. *Multipoint Correlation Functions with Finite Temperature Lanczos Method*. Bachelor’s Thesis, Technische Universität Wien. 2022.
- [18] Ian Sammis and John Strain. “A geometric nonuniform fast Fourier transform”. In: *Journal of Computational Physics* 228.18 (2009), pp. 7086–7108.
- [19] Hiroshi Shinaoka et al. “Compressing Green’s function using intermediate representation between imaginary-time and real-frequency domains”. In: *Physical Review B* 96.3, 035147 (2017).
- [20] Hiroshi Shinaoka et al. “Efficient ab initio many-body calculations based on sparse modeling of Matsubara Green’s function”. In: *SciPost Physics Lecture Notes* 63 (2022).
- [21] Hiroshi Shinaoka et al. “Overcomplete compact representation of two-particle Green’s functions”. In: *Physical Review B* 97.20, 205111 (2018).
- [22] *SimplexQuad.jl*. <https://github.com/eschnett/SimplexQuad.jl>. 2023.
- [23] *SparseIR.jl*. <https://github.com/SpM-lab/SparseIR.jl>. 2023.
- [24] John Strain. “Fast Fourier transforms of piecewise polynomials”. In: *Journal of Computational Physics* 373 (2018), pp. 346–369.
- [25] Jiachang Sun and Huiyuan Li. “Generalized Fourier transform on an arbitrary triangular domain”. In: *Advances in Computational Mathematics* 22 (2005), pp. 223–248.
- [26] Markus Wallerberger. *Fermions.jl - ED solver written in Julia*. (private communication). 2023.