**TECHNISCHE
UNIVERSITÄT
WIEN**
Vienna University of Technology

Diplomarbeit

Diagrammatic Quantum Monte Carlo with Worm Sampling

ausgeführt am

Institut für Festkörperphysik
der Technischen Universität Wien

unter der Anleitung von

Univ.Prof. Dipl.-Phys. Dr.rer.nat. Karsten Held
Dipl.-Ing. Markus Wallerberger

durch

Patrik Gunacker

_____   _____
Datum   Unterschrift

Abstract

This work focuses on methods to extract physical quantities of the quantum impurity problem, that is, among others, at the computational heart of dynamical mean field theory (DMFT). DMFT is a many-body method, which is capable of describing the physics of strongly correlated electrons. One way to tackle the impurity problem is by using Quantum Monte Carlo (QMC) impurity solvers. During the last decade, continuous-time QMC solvers became state of the art algorithms to complete this task. This work deals with the implementation of worm sampling within the hybridization expansion (CT-Hyb). Worm sampling is long known in the Monte Carlo community, but has only recently been adapted to QMC algorithms.

In Chapter 1 we give an introduction to the Hubbard model and the mapping onto the Anderson impurity model within the DMFT approximation. We then turn our focus on the derivation of the hybridization expansion. This will serve as a mathematical and physical basis for the discussion of the QMC implementation. At the end of this chapter we introduce some properties of one- and two-particle Green's functions. These two functions include almost all physics encoded in the impurity model.

In Chapter 2 we give an introduction to Monte Carlo integration. In the second half, we apply the concepts of Monte Carlo integration to the hybridization expansion derived in Chapter 1, resulting in the QMC algorithm. Lastly, we discuss the fermionic sign in CT-Hyb in more detail. As the sign problem is present in one way or another in all QMC implementations, it is important to have a good understanding on how and why it is occurring.

While Chapter 1 and 2 can be considered to be the foundation of this work, Chapter 3 builds upon these concepts to develop the theory of worm sampling in CT-Hyb. We focus on how to measure the one- and two-particle Green's function using this sampling scheme. We motivate worm sampling by pointing out the differences in the estimator with respect to sampling in partition function space. We expect better results for cases, where the estimator of partition function sampling breaks down. Further, worm sampling opens the possibility of sampling off-diagonal elements of the two-particle Green's functions.

In the last chapter of this work we will present the results of measuring Green's function using worm sampling. We will benchmark our algorithm for metallic systems and the Mott metal-insulator transition against measurements of the one-particle Green's function in partition function sampling. We further show how the worm algorithm performs for a two-orbital model with Slater-Kanamori interactions. Lastly, we present results of the two-particle Green's function using worm sampling and partition function sampling.

Zusammenfassung

Diese Arbeit befasst sich mit Methoden um physikalische Größen für das Quanten-störstellenproblem zu extrahieren, welches unter anderem die Grundlage der Dynamischen Molekularfeldnäherung (DMFT) bildet. DMFT ist eine Vielteilchen-theorie, welche in der Lage ist, die Physik stark korrelierter Elektronen zu beschreiben. Eine Möglichkeit um das Quantenstörstellenproblem zu lösen, ist die Benutzung von Quanten-Monte-Carlo (QMC)-Algorithmen. Innerhalb der letzten Jahre wurden Zeitkontinuum-QMC-Algorithmen der Stand der Technik. Diese Arbeit behandelt die Implementierung des Wurmalgorithmus in die Hybridisierungs-entwicklung (CT-Hyb). Wurmalgorithmen sind für unterschiedliche Monte-Carlo-Varianten bekannt, wurden aber erst kürzlich für QMC-Algorithmen adaptiert.

In Kapitel 1 geben wir eine Einleitung in das Hubbard-Modell und dessen Abbildung auf das Anderson-Störstellenmodell im Rahmen der DMFT. Danach befassen wir uns mit der Ableitung der Hybridisierungsentwicklung. Diese wird als mathematische und physikalische Basis für die Diskussion der QMC-Implementierung dienen. Abschließend werden einige Eigenschaften der Ein- und Zwei-Teilchen Greenschen Funktionen behandelt. Diese beiden Funktionen beinhalten beinahe die gesamte Physik des Störstellenmodells.

In Kapitel 2 geben wir eine Einleitung in die Monte-Carlo-Integration. In der zweiten Hälfte werden die Konzepte der Monte-Carlo-Integration auf die Hybridisierungsentwicklung aus Kapitel 1 angewandt, woraus der QMC-Algorithmus folgt. Abschließend behandeln wir das fermionische Vorzeichen in CT-Hyb genauer. Da das Vorzeichenproblem in allen QMC-Algorithmen auf die ein oder andere Weise vorhanden ist, ist es notwendig, ein gutes Verständnis über dessen Aufbau und Ursache zu bekommen.

Während Kaptiel 1 und 2 als Grundlage dieser Arbeit gesehen werden können, baut Kapitel 3 auf diese Konzepte auf, um die Theorie des Wurmalgorithmus zu entwickeln. Wir beschränken uns auf die Messung der Ein- und der Zwei-Teilchen Greenschen Funktionen. Wir motivieren den Wurmalgorithmus im Hinblick auf die Unterschiede der Schätzfunktion zu der Messung im Zustandssummenraum. Wir erwarten bessere Ergebnisse für Fälle, in denen die Schätzfunktion des Zustandssummenraumes ungültig wird. Außerdem erlaubt der Wurmalgorithmus Außerdiagonalelemente der Zwei-Teilchen Greenschen Funktion zu messen.

In dem letzten Kapitel liefern wir die Ergebnisse der Messung mittels Wurmalgorithmus. Wir vergleichen den Algorithmus für metallische Systeme und den Mott-Metall-Isolator-Übergang mit Messungen der Ein-Teilchen Greenschen Funktion im Zustandssummenraum. Außerdem zeigen wir Ergebnisse des Zwei-Orbital-Modells für Slater-Kanamori-Wechselwirkungen. Letztlich präsentieren wir die Ergebnisse der Zwei-Teilchen Greenschen Funktion mittels Wurmalgorithmus und der Messung im Zustandssummenraum.

Contents

1   Introduction

A physical theory is credible if it is capable of describing experimental results properly and does not contradict other theories or fundamental laws. It is common that two or more theories exist to describe phenomena on different scales. If this is the case, it is important to patch these theories together within their intermediate regions. A typical example is the coexistence of quantum theory and classical theory, both giving reliable results within different length and energy scales and the correspondence principle patching them together. While in the early days of physics we favored exact theories, which were in a sense "analytic", we have reached a point where such analytic theories are just too limited to describe the rich and detailed physics, which may well be uncovered by approximate and numerical methods. In a very famous quote this problem is addressed in the following way [1]: "In eighteenth-century Newtonian mechanics, the three-body problem was insoluble. With the birth of general relativity around 1910 and quantum electrodynamics in 1930, the two- and one-body problems became insoluble. And within modern quantum field theory, the problem of zero bodies (vacuum) is insoluble. So, if we are out after exact solutions, no bodies at all is already too many!" On the other hand, it is also well known that we can get very reasonable results of many-body systems using statistical physics. Some problems become numerically exact in the thermodynamic limit of an infinite amount of particles $N \to \infty$.

In solid state physics the quantum mechanical many-body problem is encountered when considering the large amount of electrons within the lattice of an element. Solving the electronic wave function for periodic lattices (i.e. periodic potentials) without any electron-electron interaction is trivial and is described by Bloch states. Many interesting phenomena, however, are only present due to electron-electron interactions. First attempts to solve quantum many-body problems for arbitrary potentials were made using the Hartree-Fock approximation. The Hartree-Fock approximation describes electron-electron interaction using a single Slater determinant which obeys the Pauli principle (exchange interaction). It does, however, not include Coulomb repulsion between electrons (correlation). A next step into solving the many-body problem was taken with density functional theory (DFT) [2]. DFT does not solve a quantum mechanical wave function with 3N coordinates (3 spatial coordinates times the number of particles N) but simply a wave function with 3 coordinates, namely the density of the system. DFT includes exchange and correlation. In order to approximate these two interactions, typically the local density approximation (LDA) is employed [3]. LDA is an approximation, as it assumes that the exchange and correlation DFT functional only depends on the local density. This assumption is true for a uniform electron gas, which has a constant electron density. While LDA works surprisingly well for most materials, it fails for strongly correlated materials (materials with partially filled d- or

f-orbitals). A step in describing strongly correlated materials was taken with the dynamical mean field theory (DMFT) [4]. DMFT solves the many-body problem in a self-consistency loop. As opposed to DFT, it is a true many-body theory that requires numerically expensive techniques in order to be solved. This work focuses on aspects of solving the many-body problem in dynamical mean field theory.

1.1 Hubbard Model

We are able to describe many-body physics very elegantly in second quantization. A first step in finding solutions to the many-body problems is constructing a suitable model Hamiltonian. The Hubbard model [5] is a many-body problem on a lattice, making it a good approximation for strongly correlated materials. As opposed to the tight-binding model, the Hubbard model includes an on-site Coulomb repulsion term. One of the major successes of the Hubbard model is its ability to determine the Mott-Hubbard metal insulator transition (MIT) [6]. The Hubbard Hamiltonian follows as:

$$H_{Hubbard} = U \sum_i \underbrace{d_{i\uparrow}^\dagger d_{i\uparrow}}_{n_{i\uparrow}} \underbrace{d_{i\downarrow}^\dagger d_{i\downarrow}}_{n_{i\downarrow}} - t \sum_{\langle ij \rangle, \sigma} \left(d_{i\sigma}^\dagger d_{j\sigma} + d_{j\sigma}^\dagger d_{i\sigma} \right). \tag{1.1}$$

The index i runs over all lattice sites and the index-pair $\langle ij \rangle$ runs over all nearest lattice neighbors, $\sigma \in \{\uparrow, \downarrow\}$ is the spin index, d and $d^\dagger$ denote creation and annihilation operators. The first part of the Hamiltonian leads to electronic correlation (i.e. the Coulomb repulsion U) due to double occupation on a single site. When acting on any state, an annihilation operator followed by a creation operator $d_{i\sigma}^\dagger d_{i\sigma}$ gives the occupation at this site i with spin σ. This is usually abbreviated by the density operator $n_{i\sigma}$. The second part of the Hamiltonian describes the hopping of an electron from one site i to a neighboring site j or vice versa, with a hopping amplitude $-t$. Sometimes we refer to this part as the kinetic energy term. We have presented the one-band Hubbard Hamiltonian without frustration here, that is we only consider nearest neighbors. An illustration of the Hubbard model is given in Figure 1.

We will discuss some generalizations later. It turns out that even the one-band Hubbard model is in principle impossible to solve in a straight-forward way, when i ranges over a large number of lattice sites. We point out that the Hubbard Hamiltonian is a model Hamiltonian, which we consider instead of a full ab initio many-body Hamiltonian for a given material.

In DMFT we map the many-body problem of the Hubbard model onto a single site Anderson impurity model. This mapping is exact for a lattice with infinite coordination number (i.e. an infinite number of nearest neighbors). As a result of integrating out the spatial properties of the lattice during the mapping procedure, all correlations become local [7].
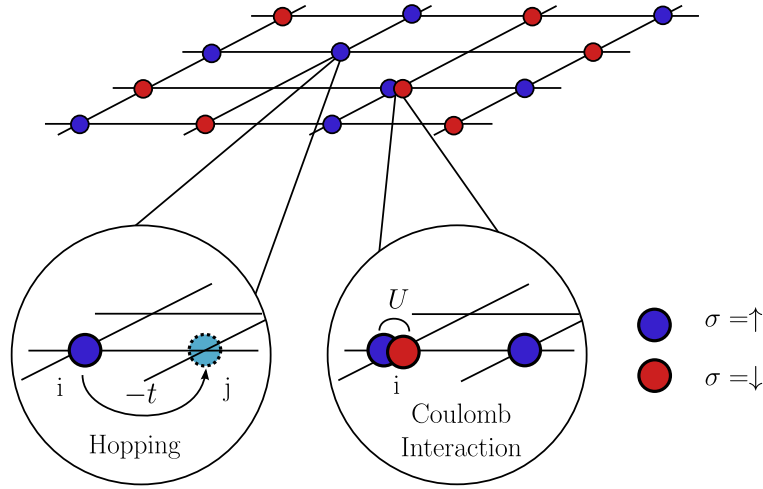
Figure 1: Illustration of the Hubbard model for a square lattice with hopping $-t$ and Coulomb repulsion U. Blue dots represent spin-up electrons and red dots spin-down electrons.
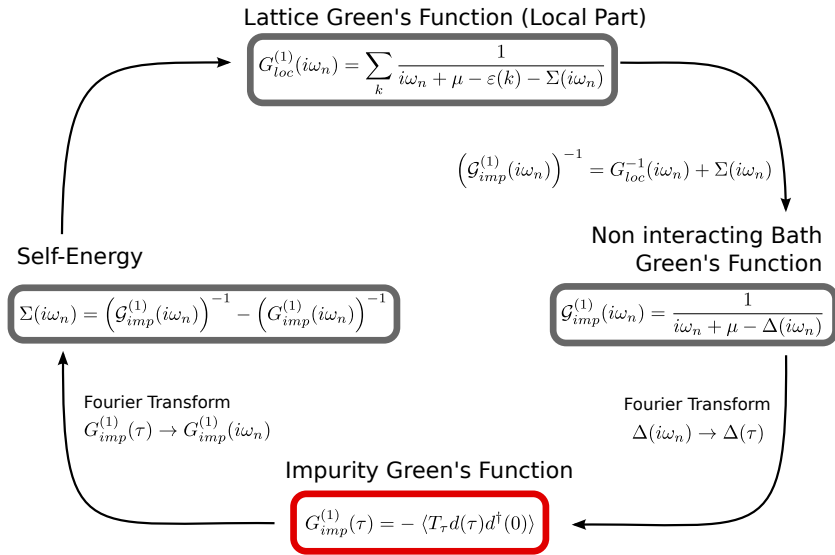
Figure 2: Illustration of the DMFT loop for the one-band case. We extract the hybridization function Δ (see Section 1.3.4) from the non-interacting bath Green's function and use it as an input to the impurity problem. From the impurity Green's function we can then extract the self-energy and calculate a new lattice Green's function. All of these steps make use of the Dyson equation in one way or another. Intermediate steps include Fourier transforms of the hybridization function and the impurity Green's function from Matsubara frequencies to imaginary time and vice versa.

We have illustrated the DMFT loop in Figure 2. The DMFT mapping takes place on the one-particle level, thus making use of the one-particle many-body Green's function $G^{(1)}$. The Green's function $G^{(1)}$ is defined for the lattice problem and for the impurity problem. The impurity problem inherits the characteristics of the lattice problem through the hybridization function $\Delta(iw_n)$, where iw_n are discrete Matsubara frequencies. Solving the impurity problem requires setting up an impurity Hamiltonian, which we will show in the following. In this work we will not consider the entire DMFT cycle, but rather concentrate on the solution of the Anderson impurity model, which is an independent problem in its own right. We can think of the DMFT mapping of the Hubbard model onto an impurity problem in the following way: for an infinite amount of neighbors all spatial correlations are lost. We are allowed to pick any lattice site as our impurity and view all surrounding sites as the bath in which the impurity is immersed.

It turns out that DMFT gives good approximations for the three dimensional Hubbard model, however it fails in describing the predominantly non-local correlations of e.g. the two dimensional Hubbard model, which makes extensions to DMFT necessary. One way to include non-local correlations on a finite length scale is to solve impurity clusters using the dynamical cluster approximation DCA [8,9]. Diagrammatic extensions, on the other hand, describe non-local correlations on all length scales. Promising methods include the dynamical vertex approximation DΓA [10], the dual fermion approach DF [11] and the one-particle irreducible approximation 1PI [12]. All these diagrammatic extensions require the calculation of two-particle quantities for the Anderson impurity model, which will also be part of this work.

1.2 Anderson Impurity Model

For extracting such two-particle quantities and other observables one needs a quantum impurity solver, which solves the Anderson impurity model. This model describes the interaction of a single impurity with a bath of non-interacting fermions [13].

1.2.1 Single-Orbital Model

In second quantization the single-orbital, single-site Anderson impurity Hamiltonian is given by:

$$H_{AIM} = U \underbrace{\underbrace{d^\dagger_\uparrow d_\uparrow}_{n_\uparrow} \underbrace{d^\dagger_\downarrow d_\downarrow}_{n_\downarrow}}_{H^I_{loc}} - \mu \underbrace{(\underbrace{d^\dagger_\uparrow d_\uparrow}_{n_\uparrow} + \underbrace{d^\dagger_\downarrow d_\downarrow}_{n_\downarrow})}_{H^0_{loc}} + \underbrace{\sum_{\vec{k},\sigma} \varepsilon_{\vec{k}\sigma} c^\dagger_{\vec{k}\sigma} c_{\vec{k}\sigma}}_{H_{bath}} + \underbrace{\sum_{\vec{k},\sigma} V_{\vec{k}\sigma} c^\dagger_{\vec{k}\sigma} d_\sigma}_{\tilde{H}_{hyb}} + \underbrace{\sum_{\vec{k},\sigma} V^*_{\vec{k}\sigma} d^\dagger_\sigma c_{\vec{k}\sigma}}_{\tilde{H}^\dagger_{hyb}},$$

$$(1.2)$$

Figure 3: Illustration of the Anderson Impurity model with Coulomb repulsion and bath hybridization. Blue dots represent spin-up electrons and red dots spin-down electrons. The non-interactive bath co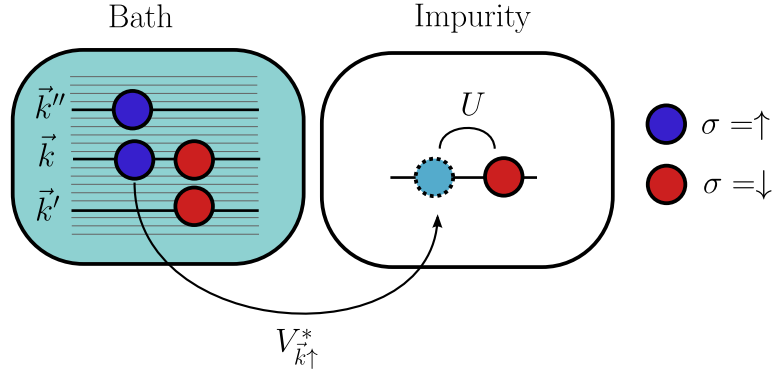nsists of continuously spread bath levels indexed by an energy k. The impurity is visualized as spin-up and spin-down orbital.

where we refer to the impurity annihilation and creation operators with d and $d^\dagger$ and to the bath annihilation and creation operators with c and $c^\dagger$. We will use this notation consistently throughout this work (one may think of 'd' being an abbreviation of "dot" as a synonym for impurity and 'c' being an abbreviation for "container" as a synonym for bath).

We separate the Anderson Hamiltonian into a part describing the impurity $H_{loc} = H_{loc}^0 + H_{loc}^I$, a part describing the non-interacting bath H_{bath} and a part describing the interaction between bath and impurity called the hybridization term $H_{hyb} = \tilde{H}_{hyb} + \tilde{H}_{hyb}^\dagger$.

Impurity Interaction Term H_{loc}^I

H_{loc}^I represents the interaction of electrons inside the impurity with one another. In the single-orbital case we only allow for density-density type interactions with a Coulomb repulsion of U:

$$H_{loc,DD}^I = \frac{1}{2} \sum_{\sigma \neq \sigma'} U n_\sigma n_{\sigma'}. \tag{1.3}$$

Later we will see a more general form of the interaction matrix $U_{\alpha\beta\gamma\delta}$, where Greek indices denote orbitals. Simplifications of the full U matrix include density-density type interactions for arbitrary orbitals (see equation (1.6)) and Slater-Kanamori type interactions (see equation (1.8)).

10

Chemical Potential Term H_{loc}^0

H_{loc}^0 represents the chemical potential contribution μ of each flavor. When talking about the "flavor" of a particle, we really mean the combined spin-orbital quantum number. We have introduced a negative sign to reproduce the analogy to the grand canonical partition function $Z = e^{-\beta(H-\mu N)}$, where β is the inverse temperature and N the number of particles in the system.

Bath Term H_{loc}^I

In this work we will solve the Anderson impurity model under the assumption of a non-interacting bath, parameterized by eigenstates $\vec{k}$ and eigenenergies $\varepsilon_{\vec{k}}$. Dealing with a non-interacting bath makes $\varepsilon_{\vec{k}}$ diagonal in flavor. We note that the operators $c_{\vec{k}}, c_{\vec{k}}^\dagger$ operate in k-space and the impurity operators $d_\sigma, d_\sigma^\dagger$ operate in real space.

Hybridization Terms $\tilde{H}_{hyb}, \tilde{H}_{hyb}^\dagger$

The hybridization terms $\tilde{H}_{hyb}, \tilde{H}_{hyb}^\dagger$ couple the impurity to the bath. Sometimes we combine these two terms to $H_{hyb} = \tilde{H}_{hyb}^\dagger + \tilde{H}_{hyb}$. The term $\tilde{H}_{hyb}^\dagger$ describes an electron taken from the bath being inserted into the impurity. $\tilde{H}_{hyb}$ describes an electron taken from the impurity being inserted back into the bath.

1.2.2 Generalizations

We have hitherto described the Hubbard Hamiltonian (1.1) and the Anderson Impurity Hamiltonian (1.2) for the single-orbital case for simplicity. In the following we will give the generalizations to an arbitrary number of bands. The Hubbard Hamiltonian generalizes to:

$$H_{Hubbard} = \frac{1}{2} \sum_i \sum_{\alpha\beta\gamma\delta} \sum_{\sigma\sigma'} U_{\alpha\beta\gamma\delta} d_{i\alpha\sigma}^\dagger d_{i\beta\sigma'}^\dagger d_{i\delta\sigma'} d_{i\gamma\sigma}$$
$$- \frac{1}{2} \sum_{\langle ij \rangle} \sum_{\alpha\beta} \sum_\sigma t_{\alpha\beta} \left(d_{i\alpha\sigma}^\dagger d_{j\beta\sigma} + d_{j\beta\sigma}^\dagger d_{i\alpha\sigma} \right), \quad (1.4)$$

where we use Greek indices $\alpha, \beta, \gamma, \delta$ as band indices. We have generalized the Coulomb interaction to $U_{\alpha\beta\gamma\delta}$ and the hopping to $-t_{\alpha\beta}$. The Anderson impurity model for a non-interacting bath generalizes to:

$$H_{AIM} = \frac{1}{2} \sum_{\alpha\beta\gamma\delta} \sum_{\sigma\sigma'} U_{\alpha\beta\gamma\delta} d^\dagger_{\alpha\sigma} d^\dagger_{\beta\sigma'} d_{\delta\sigma'} d_{\gamma\sigma} + \frac{1}{2} \sum_{\alpha\beta} \sum_{\sigma} \varepsilon_{\alpha\beta} d^\dagger_{\alpha\sigma} d_{\beta\sigma} + \sum_{\vec{k},\alpha,\sigma} \varepsilon_{\vec{k}\alpha} c^\dagger_{\vec{k}\alpha\sigma} c_{\vec{k}\alpha\sigma}$$
$$+ \sum_{\vec{k},\alpha,\sigma} V_{\vec{k}\alpha\sigma} c^\dagger_{\vec{k}\alpha\sigma} d_{\alpha\sigma} + \sum_{\vec{k},\alpha,\sigma} V^*_{\vec{k}\alpha\sigma} d^\dagger_{\alpha\sigma} c_{\vec{k}\alpha\sigma}. \quad (1.5)$$

For the most general form of the density-density interaction only orbital-diagonal elements $U_{\alpha\gamma} := U_{\alpha\beta\gamma\delta}\delta_{\alpha\beta}\delta_{\gamma\delta}$ yield a value different from 0. We usually also add additional spin-dependency to this term, such that $U_{\alpha\beta} \to U_{\alpha\sigma,\beta\sigma'}$. Then the generalization to the density-density interaction for an arbitrary number of orbitals is given by:

$$H^I_{loc,DD} = \frac{1}{2} \sum_{\alpha\beta,\sigma\sigma'} U_{\alpha\sigma,\beta\sigma'} n_{\alpha\sigma} n_{\beta\sigma'}, \quad (1.6)$$

where we require $U_{\alpha\sigma,\beta\sigma'} = U_{\beta\sigma',\alpha\sigma}$ and $U_{\alpha\sigma,\alpha\sigma} = 0$. We can simplify (1.6) by introducing a intra-orbital Hubbard U, a inter-orbital Hubbard U' and a Hund's coupling J, such that:

$$H^I_{loc,DD} = \underbrace{\sum_{\alpha} U n_{\alpha\uparrow} n_{\alpha\downarrow}}_{\text{intra-orbital}} + \underbrace{\sum_{\alpha>\beta} \sum_{\sigma} \left(U' n_{\alpha\sigma} n_{\beta,-\sigma} + (U' - J) n_{\alpha\sigma} n_{\beta\sigma} \right)}_{\text{inter-orbital}}. \quad (1.7)$$

Apart from generalizations to density-density interactions for the multi-orbital Anderson impurity Hamiltonian, we can model spin-flip and pair-hopping terms of Hund's coupling [14]. Here we only consider a orbital-diagonal hybridization. For the Slater-Kanamori interaction H^I_{loc} has the form (see e.g. References [15,16]):

$$H^I_{loc,SK} = \underbrace{\sum_{\alpha} U n_{\alpha\uparrow} n_{\alpha\downarrow}}_{\text{intra-orbital}} + \underbrace{\sum_{\alpha>\beta} \sum_{\sigma} \left(U' n_{\alpha\sigma} n_{\beta,-\sigma} + (U' - J) n_{\alpha\sigma} n_{\beta\sigma} \right)}_{\text{inter-orbital}}$$
$$- \sum_{\alpha\neq\beta} J \left(\underbrace{d^\dagger_{\alpha\downarrow} d^\dagger_{\beta\uparrow} d_{\beta\downarrow} d_{\alpha\uparrow}}_{\text{spin-flip}} + \underbrace{d^\dagger_{\beta\uparrow} d^\dagger_{\beta\downarrow} d_{\alpha\uparrow} d_{\alpha\downarrow}}_{\text{pair-hopping}} \right). \quad (1.8)$$

The algorithms developed in this work will be tested against Slater-Kanamori type interactions for the multi-orbital case. At this point we also note that, in principle, it is possible to connect several impurities to one another and solve the impurity problem for this cluster of impurities in the dynamical cluster approximation DCA [8, 9] or in Cluster DMFT [17, 18]. In this work, however, we will only focus on single-site impurities.

We will only consider quantum Monte Carlo impurity solvers to find solutions to the single-site Anderson impurity model. Other popular methods include perturbation theory, such as the iterated perturbation theory [19] or the non-crossing approximation [20] and bath discretization methods, such as exact diagonalization, matrix product states [21], numerical renormalization group [22] or density matrix renormalization group [23]! These methods require approximations to be made at one point or another, making them perform very good for a specific class of parameters, while performing poorly in other situations. Quantum Monte Carlo methods, on the other hand, are numerically exact (if we just allow for long enough simulation times) and can be used for a much broader range of parameters. In the following we will derive the physical and mathematical framework of quantum Monte Carlo solvers.

1.3 Hybridization Expansion

1.3.1 Thermal Expectation Value

In order to extract information from the Anderson impurity Hamiltonian (1.2), we have to calculate quantum mechanical expectation values. Expectation values of observables in thermal quantum field theory are given by:

$$\langle \mathcal{O}(\tau) \rangle = \frac{1}{Z} Tr \left(T_\tau e^{-\beta H} \mathcal{O}(\tau) \right) = \frac{Tr \left(T_\tau e^{-\beta H} \mathcal{O}(\tau) \right)}{Tr \left(T_\tau e^{-\beta H} \right)}, \tag{1.9}$$

where Z is the partition function and T_τ is the time ordering operator in imaginary time. In the Heisenberg picture operators evolve with $\mathcal{O}(\tau) = e^{\tau H} \mathcal{O} e^{-\tau H}$.
Equation (1.9) motivates the use of the imaginary time $\tau = -it$. Formally this relates to a Wick rotation from the real time axis to the imaginary time axis. We notice that imaginary time allows us to treat time τ and inverse temperature β on an equal footing, as the exponents $e^{\tau H}$ and $e^{\beta H}$ now cover the same domain. We will later show how quantities in imaginary times τ can be Fourier transformed to Matsubara frequencies $i\omega$. In order to extract information on the real frequency axis, we then need to perform an analytic continuation of our data from the imaginary to the real axis. This is usually one of the major drawbacks of quantum Monte Carlo algorithms, as we will see in the following. When looking at the relationship between the Green's function $G^{(1)}(\tau)$ in imaginary time and the spectral function $A(\omega)$ in real frequencies, we find:

$$G^{(1)}(\tau) = \int_{-\infty}^{\infty} d\omega \frac{e^{-\tau\omega}}{1 + e^{-\beta\omega}} A(\omega). \tag{1.10}$$

Quantum Monte Carlo algorithms have high-frequency noise usually hiding the proper asymptotics of $G^{(1)}(i\omega)$ for $i\omega \to \infty$. As a consequence of this noise and
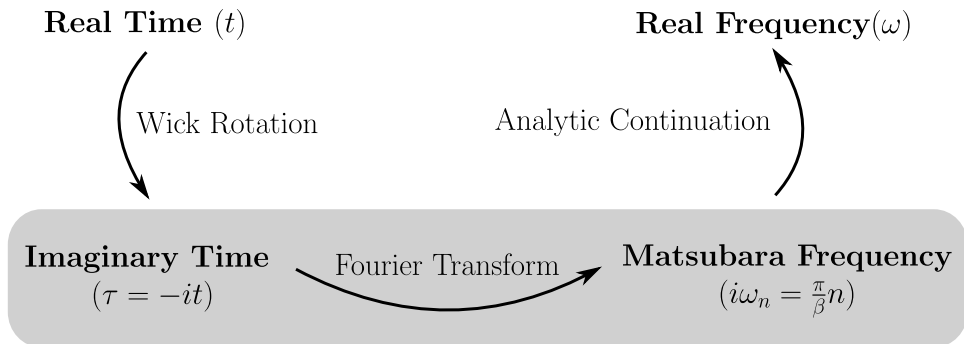
Figure 4: Relationship between real time t, imaginary time τ, discrete Matsubara frequencies $i\omega_n$ and real frequencies ω. Quantum Monte Carlo impurity solvers operate in the region of the diagram shaded in grey. The maximum entropy method allows for transformations $G(\tau) \to A(w)$ and $G(i\omega) \to A(w)$ depending on what kernel we choose.

the smallness of the integration kernel for large ω, in equation (1.10), finding one unique $A(\omega)$ to the supplied $G^{(1)}(\tau)$ is an ill-posed problem. The statistical method of choice to find the most likely $A(\omega)$ is the maximum entropy method [24].

An illustration of the transformations of real time, imaginary time, Matsubara frequencies and real frequencies as they are encountered in quantum Monte Carlo impurity solvers is given in Figure 4. As a side note we would like to point out that also real-time quantum Monte Carlo solvers exist for solving non-equilibrium problems [25]. However, these solvers still require analytic continuation as they operate on a Keldysh contour.

1.3.2 Partition Function Expansion

In the following we will expand the partition function Z in the interaction picture, where $H = H_0 + H_I$. Operators $\mathcal{O}$ then evolve with the unperturbated part H_0:

$$\mathcal{O}(\tau) = e^{\tau H_0} \mathcal{O} e^{-\tau H_0}. \tag{1.11}$$

The partition function Z is given by:

$$Z = Tr\left[T_\tau e^{-\beta H}\right] = Tr\left[T_\tau e^{-\beta H_0} e^{-\beta H_I}\right]. \tag{1.12}$$

In quantum mechanics we define the time evolution operator $U(\tau_1, \tau_2)$, which obeys:

$$\psi(\tau_1) = U(\tau_1, \tau_2)\psi(\tau_2). \tag{1.13}$$

Inserting this equation into the time-dependent Schrödinger equation results in

$U(\tau_1, \tau_2) = e^{-(\tau_1 - \tau_2)H}$. We can rewrite this expression with (1.11) in the interaction picture:

$$U_I(\tau_1, \tau_2) \equiv e^{\tau_1 H_0} U(\tau_1, \tau_2) e^{-\tau_2 H_0} = e^{\tau_1 H_0} e^{-(\tau_1 - \tau_2)H} e^{-\tau_2 H_0} \tag{1.14}$$

$$U_I(\tau) \equiv U_I(\tau, 0) = e^{\tau H_0} e^{-\tau H}. \tag{1.15}$$

We can now calculate the time derivative of the time evolution operator in the interaction picture:

$$\frac{\partial U_I(\tau)}{\partial \tau} = \frac{\partial}{\partial_\tau} \left(e^{\tau H_0} e^{-\tau H} \right)$$

$$= e^{\tau H_0} \underbrace{(H_0 - H)}_{-H_I} e^{-\tau H} = - \underbrace{e^{\tau H_0} H_I e^{-\tau H_0}}_{H_I(\tau)} e^{\tau H_0} e^{-\tau H} = -H_I(\tau) U_I(\tau). \tag{1.16}$$

The above is nothing but the Schrödinger equation for the time evolution operator $U_I(\tau)$. Integration gives the Dyson series:

$$U_I(\tau) = \mathbb{1} + \sum_{n=1}^{\infty} \frac{(-1)^n}{n!} \int_0^\tau d\tau_n \ldots \int_0^\tau d\tau_1 \, T_\tau \, H_I(\tau_n) \ldots H_I(\tau_1) = T_\tau e^{-\int_0^\tau d\tau' H_I(\tau')}. \tag{1.17}$$

We do not include $n = 0$ into the summation, as we want to stress the fact that we use $U_I(0) = \mathbb{1}$ as the boundary condition for the differential equation (1.16). We use time ordering T_τ in order to account for non-commutativity of $H_I(\tau_i), H_I(\tau_j)$ at different times. The time ordering operator sorts operators such that earlier times are to the right of later times. The difference between equation (1.15) and (1.17) is that the latter is a series up to infinite expansion order, which we can calculate numerically by introducing a manual or automatic cut-off condition. We also point out the close resemblance of (1.17) to the path integral formulation. In fact some algorithms of continuous-time quantum Monte Carlo (CT-QMC) are usually formulated in terms of path integrals.

We can now use (1.17) to rewrite the partition function (1.12) as:

$$Z = Tr \left[T_\tau e^{-\beta H_0} e^{-\beta H_I} \right]$$

$$= Tr \left[e^{-\beta H_0} \right] + \sum_{n=1}^{\infty} \frac{(-1)^n}{n!} \int_0^\beta d\tau_n \ldots \int_0^\beta d\tau_1 \, T_\tau \, Tr \left[e^{-\beta H_0} H_I(\tau_n) \ldots H_I(\tau_1) \right]. \tag{1.18}$$

1.3.3 Partition Function of the Anderson Impurity Model

Up to this point, the above derivation is a very general one valid for a large class of Hamiltonians H. The only requirement we imposed is, that it is necessary to

write H in the interaction picture $H = H_0 + H_I$. In this section we will discuss the partition function expansion with $H = H_{AIM}$. Two popular choices for H_0 and H_I are summarized in the following table:

	Interaction Expansion	Hybridization Expansion
H_0	$H_{bath} + H_{hyb} + H_{loc}^0$	$H_{bath} + H_{loc}^0 + H_{loc}^I$
H_I	H_{loc}^I	H_{hyb}

Both the hybridization expansion and the interaction expansion have different advantages and disadvantages, especially with regards to the scaling with U_{ijkl} in equation (1.5). In the CT-QMC community we refer to these expansions as CT-Hyb and CT-Int. Further techniques, such as CT-Aux, CT-J or CT-Bold exist, but will not be discussed in this work. A very detailed summary of all methods is given in Reference [26]. Instead we will focus here only on the hybridization expansion CT-Hyb. Its major advantage is that it allows for any type of interaction U_{ijkl} as $H_I = H_{hyb}$ does not depend on the interaction. We however note that since $H_I(\tau) = e^{\tau H_0} H_I e^{-\tau H_0}$, the time evolution of H_{hyb} does depend on the dimension of H_{loc}, resulting in an exponential scaling with the number of orbitals or impurity sites. The choice $H_I = H_{hyb}$ has been proposed in Reference [27] and the derivations in this work follow References [26, 27].

The partition function (1.18) for $H_I = H_{hyb}$ is given by:

$$Z = Tr\left[e^{-\beta H_{loc} + H_{bath}}\right] +$$

$$\sum_{n=1}^{\infty} \frac{(-1)^n}{n!} \int_0^\beta d\tau_n \ldots \int_0^\beta d\tau_1 \, T_\tau \, Tr\left[e^{-\beta(H_{loc}+H_{bath})} H_{hyb}(\tau_n) \ldots H_{hyb}(\tau_1)\right], \quad (1.19)$$

where $H_{loc} = H_{loc}^0 + H_{loc}^I$. We recall that in quantum mechanics a trace relates to a sandwich of operators with, in our case, many-body basis states such as:

$$Tr(\mathcal{O}) = \sum_n \langle n| \mathcal{O} |n\rangle. \quad (1.20)$$

This means that our operator $\mathcal{O}$ is not allowed to change all basis states $|n\rangle$ at once, otherwise the trace will evaluate to 0. We know that the particle number needs to be conserved. In combination with the time ordering $\tau_n > \tau_{n-1} > \ldots > \tau_1$ only even multiples of $H_{hyb}(\tau)$ can enter the partition function expansion. When looking at the combinatoric factor of the series, the denominator changes from $n!$ to $2n!$ and the alternating sign vanishes. We now substitute $H_{hyb}(\tau) = \tilde{H}_{hyb}(\tau) + \tilde{H}_{hyb}^\dagger(\tau)$. In other words, the perturbation in our partition function can stem from an electron being transferred from the bath to the impurity due to $\tilde{H}_{hyb}^\dagger$ or it can stem from

an electron being transferred from the impurity to the bath due to $\tilde{H}_{hyb}$. When ignoring the time-dependence of the operators, this results in product of the form:

$$(H_{hyb})^n = \left(\tilde{H}_{hyb} + \tilde{H}_{hyb}^\dagger\right)^n = \sum_{j=0}^n \binom{n}{j} \left(\tilde{H}_{hyb}\right)^{n-j} \left(\tilde{H}_{hyb}^\dagger\right)^j. \tag{1.21}$$

From this product only terms will add a non-zero contribution to the trace, where the amount of $\tilde{H}_{hyb}$ operators and $\tilde{H}_{hyb}^\dagger$ operators are balanced, i.e. where $n-j = j$. In this case, the binomial coefficient in equation (1.21) is usually referred to as the central binomial coefficient:

$$\binom{2n}{n} = \frac{(2n)!}{(n!)^2}. \tag{1.22}$$

We can rewrite the partition function (1.19) with $k = 2n$:

$$Z = Tr\left[e^{-\beta H_{loc} + H_{bath}}\right] + \sum_{k \in 2\mathbb{N}} \frac{1}{(k/2)!(k/2)!} \int_0^\beta d\tau_k \int_0^\beta d\tau_{k-1} \ldots \int_0^\beta d\tau_2 \int_0^\beta d\tau_1 \times$$

$$T_\tau Tr\left[e^{-\beta(H_{loc} + H_{bath})} \tilde{H}_{hyb}(\tau_k)\tilde{H}_{hyb}^\dagger(\tau_{k-1}) \ldots \tilde{H}_{hyb}(\tau_2)\tilde{H}_{hyb}^\dagger(\tau_1)\right]. \tag{1.23}$$

Now k relates to the total number of $\tilde{H}_{hyb}$ and $\tilde{H}_{hyb}^\dagger$ operators in the trace and $k/2$ relates to the expansion order of the partition function. In the following we will restrict our discussion to the single-orbital Anderson impurity Hamiltonian of equation (1.2). Using $\tilde{H}_{hyb} = \sum_{\vec{k}\sigma} V_{\vec{k}\sigma} c_{\vec{k}\sigma}^\dagger d_\sigma$ and $\tilde{H}_{hyb}^\dagger = \sum_{\vec{k}\sigma} V_{\vec{k}\sigma}^* d_\sigma^\dagger c_{\vec{k}\sigma}$ we now insert creation and annihilation operators for the bath and the impurity explicitly:

$$Z = Tr\left[e^{-\beta H_{loc} + H_{bath}}\right] + \sum_{k \in 2\mathbb{N}} \frac{1}{(k/2)!(k/2)!} \int_0^\beta d\tau_k \int_0^\beta d\tau_{k-1} \ldots \int_0^\beta d\tau_2 \int_0^\beta d\tau_1$$

$$\sum_{\vec{k}_k\sigma_k} \ldots \sum_{\vec{k}_1\sigma_1} V_{\vec{k}_k\sigma_k} V_{\vec{k}_{k-1}\sigma_{k-1}}^* \ldots V_{\vec{k}_2\sigma_2} V_{\vec{k}_1\sigma_1}^* \ T_\tau \ Tr\left[e^{-\beta(H_{loc} + H_{bath})}\right.$$

$$\left. c_{\vec{k}_k\sigma_k}^\dagger(\tau_k)d_{\sigma_k}(\tau_k)d_{\sigma_{k-1}}^\dagger(\tau_{k-1})c_{\vec{k}_{k-1}\sigma_{k-1}}(\tau_{k-1}) \ldots c_{\vec{k}_2\sigma_2}^\dagger(\tau_2)d_{\sigma_2}(\tau_2)d_{\sigma_1}^\dagger(\tau_1)c_{\vec{k}_1\sigma_1}(\tau_1)\right]. \tag{1.24}$$

The time evolution of the bath and impurity creation and annihilation operators in the interaction picture follows equation (1.11), where the operator in the exponent is $H_0 = H_{loc} + H_{bath}$. As the time evolution does not include the hybridization, the bath and the impurity are no longer coupled and we can split the trace:

$$Z = Tr\left[e^{-\beta H_{loc}+H_{bath}}\right] + \sum_{k\in 2\mathbb{N}} \frac{1}{(k/2)!(k/2)!} \int_0^\beta d\tau_k \int_0^\beta d\tau_{k-1} \ldots \int_0^\beta d\tau_2 \int_0^\beta d\tau_1 \times$$

$$\sum_{\vec{k}_k\sigma_k} \ldots \sum_{\vec{k}_1\sigma_1} V_{\vec{k}_k\sigma_k} V^*_{\vec{k}_{k-1}\sigma_{k-1}} \ldots V_{\vec{k}_2\sigma_2} V^*_{\vec{k}_1\sigma_1} \times$$

$$Tr_d \underbrace{\left[T_\tau e^{-\beta H_{loc}} d_{\sigma_k}(\tau_k) d^\dagger_{\sigma_{k-1}}(\tau_{k-1}) \ldots d_{\sigma_2}(\tau_2) d^\dagger_{\sigma_1}(\tau_1)\right]}_{\text{local trace}} \times$$

$$Tr_c \underbrace{\left[T_\tau e^{-\beta H_{bath}} c^\dagger_{\vec{k}_k\sigma_k}(\tau_k) c_{\vec{k}_{k-1}\sigma_{k-1}}(\tau_{k-1}) \ldots c^\dagger_{\vec{k}_2\sigma_2}(\tau_2) c_{\vec{k}_1\sigma_1}(\tau_1)\right]}_{\text{bath trace}}. \quad (1.25)$$

We will find that the local trace $T_d(\ldots)$ is difficult to calculate and requires numerical solving techniques such as QMC, while the bath trace $T_c(\ldots)$ can be rewritten into a matrix determinant, which is relatively simple to solve.

1.3.4 Bath Trace and Hybridization Matrix

In the following we will show how the bath trace can be rewritten. We include the hybridization amplitudes V, V^* into the bath trace. When combining the bath trace and the hybridization amplitudes to one expression, we are left with the hybridization matrix. First we will introduce the bath partition function:

$$Z_{bath} = Tr\left[T_\tau e^{-\beta H_{bath}}\right], \quad (1.26)$$

with $H_{bath} = \sum_{\vec{k}\sigma} \varepsilon_{\vec{k}} n_{\vec{k}\sigma}$. When assuming a non-interacting bath, the calculation of the bath partition function is straight-forward, since each bath state $|\vec{k}\sigma\rangle$ is simply empty or singly occupied ($n_{\vec{k}\sigma} = 0, 1$):

$$Z_{bath} = Tr\left[T_\tau e^{-\beta H_{bath}}\right]$$
$$= \prod_{\vec{k}\sigma}\left(\langle\vec{k}\sigma| e^{-\beta\varepsilon_{\vec{k}}\cdot 0} |\vec{k}\sigma\rangle + \langle\vec{k}\sigma| e^{-\beta\varepsilon_{\vec{k}}\cdot 1} |\vec{k}\sigma\rangle\right)$$
$$= \prod_{\vec{k}\sigma}\left(1 + e^{-\beta\varepsilon_{\vec{k}}}\right). \quad (1.27)$$

We will now take a close look at the combined factor of hybridization amplitudes $V_{k\sigma}$ and the bath trace in equation (1.25). We modify this factor with the bath

partition function Z_{bath}:

$$\frac{1}{Z_{bath}} \sum_{\vec{k}_k \sigma_k} \cdots \sum_{\vec{k}_1 \sigma_1} V_{\vec{k}_k \sigma_k} V^*_{\vec{k}_{k-1}\sigma_{k-1}} \cdots V_{\vec{k}_2 \sigma_2} V^*_{\vec{k}_1 \sigma_1} \times$$

$$Tr_c \left[T_\tau e^{-\beta H_{bath}} c^\dagger_{\vec{k}_k \sigma_k}(\tau_k) c_{\vec{k}_{k-1}\sigma_{k-1}}(\tau_{k-1}) \ldots c^\dagger_{\vec{k}_2 \sigma_2}(\tau_2) c_{\vec{k}_1 \sigma_1}(\tau_1) \right]. \quad (1.28)$$

It is instructive to take a close look what an operator pair $c^\dagger_{\vec{k}_i \sigma_i}(\tau_i), c_{\vec{k}_j \sigma_j}(\tau_j)$ does inside the bath trace. The value $Tr_c \left[T_\tau e^{-\beta H_{bath}} c^\dagger_{\vec{k}_i \sigma_i}(\tau_i) c_{k_j \sigma_j}(\tau_j) \right]$ is only non-zero if the quantum numbers are obeyed, i.e. $\vec{k}_i = \vec{k}_j$ and $\sigma_i = \sigma_j$ for some i, j, because the bath is diagonal in its eigenstates $\vec{k}$ and spin σ. Due to the Pauli principle we can assume that every bath level and spin is only occupied by zero or one electron for the interval $\tau = [0, \beta]$, where $\tau = \tau_i - \tau_j$:

$$Tr_c \left[T_\tau e^{-\beta H_{bath}} c^\dagger_{\vec{k}\sigma}(\tau_i) c_{\vec{k}\sigma}(\tau_j) \right] = \begin{cases} -e^{-\varepsilon_{\vec{k}}\tau}, & 0 < \tau < \beta \\ e^{-\varepsilon_{\vec{k}}(\tau+\beta)}, & -\beta < \tau < 0 \end{cases} \quad (1.29)$$

We now introduce the anti-periodic hybridization function $\Delta_{\sigma_i \sigma_j}$ with:

$$\Delta_{\sigma_i \sigma_j}(\tau) = \sum_{\vec{k}} \frac{V^*_{\vec{k}\sigma_i} V_{\vec{k}\sigma_j}}{1 + e^{-\beta \varepsilon_{\vec{k}}}} \times \begin{cases} -e^{-\varepsilon_{\vec{k}}\tau}, & 0 < \tau < \beta \\ e^{-\varepsilon_{\vec{k}}(\tau+\beta)}, & -\beta < \tau < 0 \end{cases} \quad (1.30)$$

We point out that $V_{k\sigma_i}$ and $V^*_{k\sigma_j}$ do not factorize to V_k and V^*_k, as we still allow off-diagonal terms for the spin σ in the local trace. We find that equation (1.28) can be written as a product of hybridization functions $\Delta_{\sigma_i \sigma_j}(\tau_i - \tau_j)$. In order to include the permutation sign due to the time ordering operator T_τ properly, we write the product of hybridization functions as a determinant rather than a permanent, such that:

$$\frac{1}{Z_{bath}} \sum_{\vec{k}_k, \sigma_k} \cdots \sum_{\vec{k}_1 \sigma_1} V_{\vec{k}_k} V^*_{\vec{k}_{k-1}\sigma_{k-1}} \cdots V_{\vec{k}_2 \sigma_2} V^*_{\vec{k}_1 \sigma_1} \times$$

$$Tr_c \left[T_\tau e^{-\beta H_{bath}} c^\dagger_{\vec{k}_k \sigma_k}(\tau_k) c_{\vec{k}_{k-1}\sigma_{k-1}}(\tau_{k-1}) \ldots c^\dagger_{\vec{k}_2 \sigma_2}(\tau_2) c_{\vec{k}_1 \sigma_1}(\tau_1) \right] = \det \mathbf{\Delta}. \quad (1.31)$$

The hybridization function connects annihilation operators to creation operators of the bath and the impurity. The matrix $\mathbf{\Delta}$ is defined with elements $\Delta_{\sigma_i \sigma_j}(\tau_i - \tau_j)$ using equation (1.30). The partition function in the hybridization expansion can now be written as:

$$Z = Tr\left[e^{-\beta H_{loc} + H_{bath}}\right] +$$
$$Z_{bath} \sum_{k \in 2\mathbb{N}} \int_{\tau_{k-1}}^{\beta} d\tau_k \int_{\tau_{k-2}}^{\beta} d\tau_{k-1} \ldots \int_{\tau_1}^{\beta} d\tau_2 \int_0^{\beta} d\tau_1 \sum_{\sigma_k} \ldots \sum_{\sigma_1} \times$$
$$Tr_d\left[e^{-\beta H_{loc}} d_{\sigma_k}(\tau_k) d_{\sigma_{k-1}}^{\dagger}(\tau_{k-1}) \ldots d_{\sigma_2}(\tau_2) d_{\sigma_1}^{\dagger}(\tau_1)\right] \times \det \boldsymbol{\Delta}. \quad (1.32)$$

In this expression, we have explicitly enforced the time-ordering with $\tau_k > \tau_{k-1} > \ldots > \tau_2 > \tau_1$. This way the integration bounds are adapted and we lose the factor $\frac{1}{(k/2)!(k/2)!}$ because there are $(k/2)!$ ways to arrange the creation operators and $(k/2)!$ ways to arrange the annihilation operators. This is the starting point for efficient algorithms using quantum Monte Carlo (QMC) techniques. Here we have assumed the single-orbital case. The generalization to the multi-orbital case is straight-forward and gives us a hybridization function of the type $\Delta_{\alpha_i \alpha_j}(\tau_i - \tau_j)$, where the α are generalized spin-orbital indices. Further, the above formula becomes much easier if we assume an orbital-diagonal hybridization function, i.e. $\Delta_{\alpha_i \alpha_i}(\tau_i - \tau_j)$, where we only insert operator pairs of the same flavor into the local trace. This way our hybridization matrix $\boldsymbol{\Delta}$ has a block diagonal form and the determinant of $\boldsymbol{\Delta}$ breaks down into a product of determinants of the sub-blocks for each orbital.

1.4   Green's Functions

Up to this point we have treated the expansion of the partition function, bringing it into a form, which can be calculated numerically using equation (1.32). We recall that we defined the thermal expectation value in equation (1.9). In principle it would be necessary to derive the hybridization expansion (or any other expansion) not only for the partition function, but for the expectation value of any local operator we want to calculate. However, the trace of a local operator and the trace of the partition function have the same structure. For the partition function the local trace in the hybridization expansion is given by:

$$Tr_d[Z] = Tr_d\left[T_\tau e^{-\beta H_{loc}} d_{\alpha_k}(\tau_k) d_{\alpha_{k-1}}^{\dagger}(\tau_{k-1}) \ldots d_{\alpha_2}(\tau_2) d_{\alpha_1}^{\dagger}(\tau_1)\right]. \quad (1.33)$$

The local trace for an operator in the hybridization expansion is instead given by:

$$Tr_d[\mathcal{O}(\tau_1, \ldots \tau_i, \ldots)] =$$
$$Tr_d\left[T_\tau e^{-\beta H_{loc}} d_{\alpha_k}(\tau_k) d_{\alpha_{k-1}}^{\dagger}(\tau_{k-1}) \ldots d_{\alpha_2}(\tau_2) d_{\alpha_1}^{\dagger}(\tau_1) \mathcal{O}(\tau_1, \ldots \tau_i, \ldots)\right]. \quad (1.34)$$

We observe a great similarity between equations (1.33) and (1.34). The local creation and annihilation operators $d^\dagger, d$ are a result of treating the factor $e^{-\beta H_{hyb}}$ perturbatively. As suggested in equation (1.34) it is very tempting to just insert any local operator $\mathcal{O}(\tau_1, \ldots \tau_i, \ldots)$ into the local trace and calculate it to find the expectation value up to a normalization factor. Following this route is actually the main topic of this work, where the corresponding algorithm of the QMC solver is referred to as worm sampling. But let us take a step back and discuss the operator $\mathcal{O}(\tau_1, \ldots \tau_i, \ldots)$ more carefully. A very specific case might be that we choose $\mathcal{O}(\tau_i, \tau_j) = d(\tau_i)d^\dagger(\tau_j)$. The advantage of calculating the expectation value of this operator, is that operator pairs $d(\tau_i)d^\dagger(\tau_j)$ already exist in the local trace of the partition function, only with a hybridization line connected. We would just need to cut these lines and then immediately arrive at a result for $\langle \mathcal{O}(\tau_i, \tau_j)\rangle = \langle d(\tau_i)d^\dagger(\tau_j)\rangle$ (again up to a normalization factor). In the same fashion it would also be possible to calculate the expectation value of an operator of the type $\mathcal{O}(\tau_1, \tau'_1, \ldots, \tau_n, \tau'_n) = d(\tau_1)d^\dagger(\tau'_1) \ldots d(\tau_n)d^\dagger(\tau'_n)$, where creation and annihilation operators appear pairwise. Taking the expectation value of this operator is nothing but the n-particle Green's function:

$$G^{(n)}_{\alpha_1\alpha_{1'}\ldots\alpha_n\alpha_{n'}}(\tau_1, \tau_{1'} \ldots \tau_n, \tau_{n'}) = (-1)^n \langle T_\tau d_{\alpha_1}(\tau_1)d^\dagger_{\alpha_{1'}}(\tau'_1) \ldots d_{\alpha_n}(\tau_n)d^\dagger_{\alpha_{n'}}(\tau'_n)\rangle. \tag{1.35}$$

The similarities between Green's functions and the partition function make it very intuitive to calculate expectation values in the hybridization expansion by cutting hybridization lines [26, 27]. As already indicated in this work we will take the alternative approach of explicitly inserting the operators into the trace. We will motivate this approach in Chapter 3.

1.4.1 Properties of Green's Functions

In the following we will give a short summary of some properties of Green's functions. While these properties are in general true for any Green's function $G^{(n)}$, we will reduce our discussion to the one-particle Green's function $G^{(1)}_{\alpha_1\alpha_2}$ and the two-particle Green's function $G^{(2)}_{\alpha_1\alpha_2\alpha_3\alpha_4}$, as these two functions are of central importance to this work.

Time Translation and Time Interval

If the Hamiltonian H is time independent, then the Green's function becomes translational invariant in its imaginary time argument [28]. This allows us to set the time origin of the Green's function. One choice may be to shift all operators by τ_4, such that:

$$G^{(2)}_{\alpha_1\alpha_2\alpha_3\alpha_4}(\tau_1, \tau_2, \tau_3, \tau_4) = \langle T_\tau d_{\alpha_1}(\tau_1) d^\dagger_{\alpha_2}(\tau_2) d_{\alpha_3}(\tau_3) d^\dagger_{\alpha_4}(\tau_4) \rangle \tag{1.36}$$

$$= \langle T_\tau d_{\alpha_1}(\tau_1 - \tau_4) d^\dagger_{\alpha_2}(\tau_2 - \tau_4) d_{\alpha_3}(\tau_3 - \tau_4) d^\dagger_{\alpha_4}(\tau_4 - \tau_4) \rangle \tag{1.37}$$

$$= \langle T_\tau d_{\alpha_1}(\tau_1') d^\dagger_{\alpha_2}(\tau_2') d_{\alpha_3}(\tau_3') d^\dagger_{\alpha_4}(0) \rangle \tag{1.38}$$

$$=: G^{(2)}_{\alpha_1\alpha_2\alpha_3\alpha_4}(\tau_1', \tau_2', \tau_3'). \tag{1.39}$$

Of course the same argument also holds for the one-particle Green's function, where we find:

$$G^{(1)}_{\alpha_1\alpha_2}(\tau_1, \tau_2) = -\langle T_\tau d_{\alpha_1}(\tau_1) d^\dagger_{\alpha_2}(\tau_2) \rangle \tag{1.40}$$

$$= -\langle T_\tau d_{\alpha_1}(\tau_1 - \tau_2) d^\dagger_{\alpha_2}(\tau_2 - \tau_2) \rangle \tag{1.41}$$

$$= -\langle T_\tau d_{\alpha_1}(\tau) d^\dagger_{\alpha_2}(0) \rangle \tag{1.42}$$

$$=: G^{(1)}_{\alpha_1\alpha_2}(\tau). \tag{1.43}$$

When we introduced the concept of imaginary time along equation (1.9), we found that there exists a connection between the imaginary time τ and the inverse temperature β. We can establish this link by taking advantage of the Lehmann representation. We follow the reasoning of [28], define our Hamiltonian H in its eigenbasis $|m\rangle$ with energies E_m and obtain:

$$G^{(2)}_{\alpha_1\alpha_2\alpha_3\alpha_4}(\tau_1, \tau_2, \tau_3, \tau_4) \tag{1.44}$$

$$= \frac{1}{Z} \sum_m \langle m| e^{-\beta H} d_{\alpha_1}(\tau_1) d^\dagger_{\alpha_2}(\tau_2) d_{\alpha_3}(\tau_3) d^\dagger_{\alpha_4}(\tau_4) |m\rangle \tag{1.45}$$

$$= \frac{1}{Z} \sum_m \langle m| e^{-(\beta-\tau_1)H} d_{\alpha_1} e^{(\tau_2-\tau_1)H} d^\dagger_{\alpha_2} e^{(\tau_3-\tau_2)H} d_{\alpha_3} e^{(\tau_4-\tau_3)H} d^\dagger_{\alpha_4} e^{-\tau_4 H} |m\rangle \tag{1.46}$$

$$= \frac{1}{Z} \sum_m e^{-(\beta-\tau_1+\tau_4)E_m} \langle m| d_{\alpha_1} e^{(\tau_2-\tau_1)H} d^\dagger_{\alpha_2} e^{(\tau_3-\tau_2)H} d_{\alpha_3} e^{(\tau_4-\tau_3)H} d^\dagger_{\alpha_4} |m\rangle . \tag{1.47}$$

Notice that we have left out the time ordering symbol T_τ; this means we have imposed a fixed ordering of the form $\tau_1 > \tau_2 > \tau_3 > \tau_4$. In general the Hamiltonian H is unbound from above, but bound from below. We would like to deal with a converging sum in the above equation, which means that we require the following for the exponent:

$$\beta - \tau_1 + \tau_4 > 0. \tag{1.48}$$

Combining this with the time ordering, we find:

$$\beta + \tau_4 > \tau_1 > \tau_2 > \tau_3 > \tau_4. \tag{1.49}$$

We can now exploit the time translational property from above, allowing us to shift this inequality and set $\tau_4 = 0$, such that:

$$\beta > \tau_1 > \tau_2 > \tau_3 > 0. \tag{1.50}$$

This means that the time argument of the Green's function is limited to an interval $\tau_i \in [0, \beta]$. Notice that exploiting the time translational property in the last step was only done out of convenience, but is not strictly required here. One can further derive the anti-periodicity of the Green's function using the cyclic properties of the trace and the algebra of the fermionic operators. We find:

$$G^{(1)}_{\alpha_1\alpha_2}(\tau) = -G^{(1)}_{\alpha_1\alpha_2}(\tau + \beta) \quad \text{if} \quad \tau < 0 \tag{1.51}$$

$$G^{(2)}_{\alpha_1\alpha_2\alpha_3\alpha_4}(\tau_1, \tau_2, \tau_3) = -G^{(2)}_{\alpha_1\alpha_2\alpha_3\alpha_4}(\tau_1 - \beta, \tau_2, \tau_3) \quad \text{if} \quad \tau_1 - \beta > \tau_3. \tag{1.52}$$

Fourier Transform

It is often better to take the Fourier transform of the time argument of a Green's function in order to obtain Matsubara frequencies. For $G^{(1)}_{\alpha_1\alpha_2}(\tau)$, the Fourier transform simply becomes:

$$G^{(1)}_{\alpha_1\alpha_2}(i\nu) = \int_0^\beta d\tau e^{i\nu\tau} G^{(1)}_{\alpha_1\alpha_2}(\tau). \tag{1.53}$$

The fermionic Matsubara frequencies ν_n are defined as:

$$\nu_n = \frac{\pi}{\beta}(2n+1) \qquad n \in \mathbb{Z}. \tag{1.54}$$

We can restrict ourselves to the fermionic frequencies (odd multiples of π/β) due to the anti-periodicity of the Green's function $G^{(1)}_{\alpha_1\alpha_2}(\tau)$. We note that since the time domain is bounded, the inverse transformation is not defined as a continuous Fourier transform, but rather a discrete transform of the type:

$$G^{(1)}_{\alpha_1\alpha_2}(\tau) = \frac{1}{\beta} \sum_n d\tau e^{-i\nu\tau} G^{(1)}_{\alpha_1\alpha_2}(i\nu). \tag{1.55}$$

The Fourier transform of the two-particle Green's function $G^{(2)}_{\alpha_1\alpha_2\alpha_3\alpha_4}(\tau_1, \tau_2, \tau_3, \tau_4)$ can be written correspondingly as:

$$G^{(2),ph}_{\alpha_1\alpha_2,\alpha_3,\alpha_4}(i\nu, i\nu', i\omega) =$$
$$\int_0^\beta d\tau_1 \int_0^\beta d\tau_2 \int_0^\beta d\tau_3 \int_0^\beta d\tau_4 e^{i\nu(\tau_1-\tau_2)} e^{-i\nu'(\tau_3-\tau_4)} e^{i\omega(\tau_1-\tau_4)} G^{(2)}_{\alpha_1\alpha_2\alpha_3\alpha_4}(\tau_1, \tau_2, \tau_3, \tau_4). \tag{1.56}$$

Here, we have introduced the bosonic Matsubara frequencies ω_n, which are defined as even multiples of π/β:

$$\omega_n = \frac{2\pi}{\beta}n \qquad n \in \mathbb{Z}. \tag{1.57}$$

For now we will not explain the deeper meaning of the convention of the time differences chosen in equation (1.56). We refer to this convention as the particle-hole convention. Another popular choice is the particle-particle convention. These conventions are applicable for the single-orbital case. The back-transformation of a Green's function in Matsubara frequencies works over a discrete Fourier transform of the type:

$$G^{(2)}_{\alpha_1\alpha_2,\alpha_3,\alpha_4}(\tau_1, \tau_2, \tau_3, \tau_4) =$$
$$\frac{1}{\beta^2} \sum_{\nu,\nu',\omega} e^{-i\nu(\tau_1-\tau_2)} e^{i\nu'(\tau_3-\tau_4)} e^{-i\omega(\tau_1-\tau_4)} G^{(2),ph}_{\alpha_1\alpha_2\alpha_3\alpha_4}(i\nu, i\nu', i\omega). \tag{1.58}$$

The one-particle Green's function $G^{(1)}_{\alpha_1\alpha_2}(i\nu)$ is a major ingredient of DMFT, the two-particle Green's function $G^{(2)}_{\alpha_1\alpha_2\alpha_3\alpha_4}(i\nu, i\nu', i\omega)$ is a major ingredient of the diagrammatic extensions to DMFT. A full treatment of the properties and importance of Green's function in the context of many-body physics can be found in e.g. References [28,29]. This work will focus on how to measure these two quantities using a QMC solver.

2 Monte Carlo Integration

The mapping of the Hubbard model (1.1) onto the Anderson impurity model (1.2) in DMFT and the expansion of the partition function (1.32) are first steps in solving the many-body problem. We are however still confronted with actually calculating the perturbative series of the expectation values of the operators under consideration. It turns out that solving the local trace of the expansion is the most challenging task. Let us give a short reasoning why it is not trivial to solve the local trace of our quantum impurity. The local trace represents the dynamics of the quantum impurity with a general interaction. Depending on the expansion order, the local trace is constructed by a specific amount of annihilation and creation operators. In the hybridization expansion, the annihilator and the creation operator appear pairwise. While it may seem counter-intuitive it turns out that the expansion order is generally higher for low temperatures, i.e. the trace has more operators in it. This is because the length of the trace scales with the inverse temperature β. Typically, the expansion order can go up to a few hundreds. Although we are dealing with a single site impurity (or sometimes a few cluster

sites), we encounter integrals of high dimensions due to this expansion order. In this section we will focus on how to solve such integrals.

2.1 Basic Concepts

In order to solve high dimensional integrals, such as the partition function in equation (1.32), it is common to employ Monte Carlo techniques. These methods became more powerful with advancements in computer development [30]. In the following we will give a general introduction into Monte Carlo sampling. The theory presented in this section is mostly self-contained and can be read without the previous sections. The theory in this section is based on References [31, 32]. Once we established the basic tools necessary to treat high dimensional integrals, we adapt them to our quantum impurity problem in Section 2.2.

In equation (1.9) we have introduced the thermal expectation value using the partition function $Z = Tr\left[T_\tau e^{-\beta H}\right]$. In fact the expectation value is a very general concept, not originating from quantum mechanics. In probability theory the expectation value is given by:

$$\langle A \rangle = \frac{\int_V A(x)p(x)dx}{\int_V p(x)dx}, \tag{2.1}$$

where $A(x)$ is our observable evaluated at the value of x and $p(x)$ is a (not necessarily normalized) probability density function. In order to normalize the expectation value, we divide the integral $\int_V A(x)p(x)dx$ by the integral $\int_V p(x)dx$.

Of course the above expectation value is not restricted to just one dimension x but may be formed over an arbitrary number of dimensions x_i, such that:

$$\langle A \rangle = \frac{\int_V A(x_i)p(x_i)dx_i}{\int_V p(x_i)dx_i} \qquad i \in \mathbb{N}. \tag{2.2}$$

In the following we are interested in finding the expectation values numerically. We will focus on the one dimensional case (2.1) for now. In general, one can think of two ways of calculating an integral numerically. The first is to cut the integration domain into slices Δx_i (not necessary of equal length) and sum up the area of the sub-slices. When taking the limit $\Delta x_i \to 0$, we will find the exact value of the integral. The method we will use is based on the mean value theorem for integration, which states that there exists an intermediate point $\xi \in V$ such that:

$$\int_V f(x)dx = f(\xi)V, \tag{2.3}$$

where the continuous function $f(x)$ evaluated at ξ multiplied with the integration domain gives the integral value. We will find that $f(\xi) = \langle f \rangle$ is nothing but the

expectation value. The basic idea of Monte Carlo is to sample the integral as an infinite sum with a large number N of random values from the integration domain V:

$$\int_V f(x)dx = \frac{1}{N}\sum_j^N f(x_j)V + \mathcal{O}\left(\frac{1}{N^a}\right),$$ (2.4)

where for now our random variables x_j are spread uniformly in the integration domain. We have not yet specified the error $\mathcal{O}\left(\frac{1}{N^a}\right)$, i.e. the exponent of a, but we will do so in the next section. The expectation value can be written as:

$$\langle A \rangle = \frac{\frac{V}{N}\sum_j^N A(x_j)p(x_j)}{\frac{V}{N}\sum_j^N p(x_j)} + \mathcal{O}\left(\frac{1}{N^a}\right),$$ (2.5)

where we will refer to the first term as the Monte Carlo expectation value, ignoring the approximation error:

$$\langle A \rangle_{MC} = \frac{\frac{V}{N}\sum_j^N A(x_j)p(x_j)}{\frac{V}{N}\sum_j^N p(x_j)}.$$ (2.6)

2.1.1 Uniformly Distributed Random Numbers in V

We will now consider the implications of a uniform probability density function $p(x)$:

$$p(x) = \text{const} = \frac{1}{V}.$$ (2.7)

We find:

$$\langle A \rangle_{MC} = \frac{1}{N}\sum_j^N A(x_j).$$ (2.8)

For $p(x) = \frac{1}{V}$ the denominator of our Monte Carlo expectation value becomes unity. This is of course nothing but the normalization of the probability density function. Equation (2.8) is simply the arithmetic mean of A. Now we can calculate the behavior of the approximation error defined in equation (2.5). The generalization of the error to arbitrary probability density functions $p(x)$ is not worked out at this point. We find:

$$\varepsilon_N = \langle A \rangle - \langle A \rangle_{MC} = \mathcal{O}\left(\frac{1}{N^a}\right).$$ (2.9)

We further calculate the variance of the Monte Carlo expectation value with:

$$\sigma^2_{MC}(A) = \langle A^2 \rangle_{MC} - \langle A \rangle^2_{MC}.$$ (2.10)

The central limit theorem states, that for an a set of independent, identically distributed random numbers from a probability distribution $p(x)$ the error $\sqrt{N}\varepsilon_N$ approaches a normal distribution with variance $\sigma^2_{MC}(A)$:

$$\sqrt{N}\varepsilon_N \rightarrow \frac{1}{\sqrt{2\pi}\sigma_{MC}(A)}e^{-\frac{1}{2}\left(\frac{x}{\sigma_{MC}(A)}\right)^2}. \tag{2.11}$$

The probability distribution of ε_N follows a Gaussian distribution with a variance of:

$$\sigma^2_{\varepsilon_N}(A) = \frac{\sigma^2_{MC}(A)}{N}. \tag{2.12}$$

We have now determined the exponent of the Monte Carlo error $\mathcal{O}\left(\frac{1}{N^a}\right)$ to $a = \frac{1}{2}$. This means, that if we want to reduce the Monte Carlo error by a factor of 10, we have to increase our samples by a factor of 100. This allows us to solve any integral using the Monte Carlo expectation value and the variance with:

$$\int_V A(x)dx = \left(\langle A \rangle_{MC} \pm \frac{1}{\sqrt{N}}\sigma_{MC}(A)\right)V. \tag{2.13}$$

In low dimensions there exist superior integration methods, with a better error convergence. For dimensions $d > 5$ usually Monte Carlo integration becomes the method of choice. With regards to calculating the expectation value of the partition function and local operators, where dimensions of $d \sim 100$ are likely, we are required to use Monte Carlo techniques. We also point out that Monte Carlo methods parallelize trivially, as we are allowed to calculate the integral independently on different threads and then calculate the final result as the mean value with standard deviation from the given data. It is also important to note, that the Monte Carlo error is just an estimate of the real error, as it is not guaranteed that the Monte Carlo error follows a normal distribution.

2.1.2 Importance Sampling

It is possible to reduce the variance, not only by higher statistics, but by a good choice of random configurations x_j. We usually refer to this method of variance reduction as importance sampling [31]. In the previous paragraph we assumed $p(x) = \frac{1}{V}$ being a uniform probability density function from which we choose our random variables. Many times we are confronted with a general probability density function $p(x)$ in the expectation value. We want to keep the freedom of choosing our random variables from any probability density function $g(x)$, not necessarily requiring $g(x) = p(x)$. Then we can re-weigh the integral of the expectation value
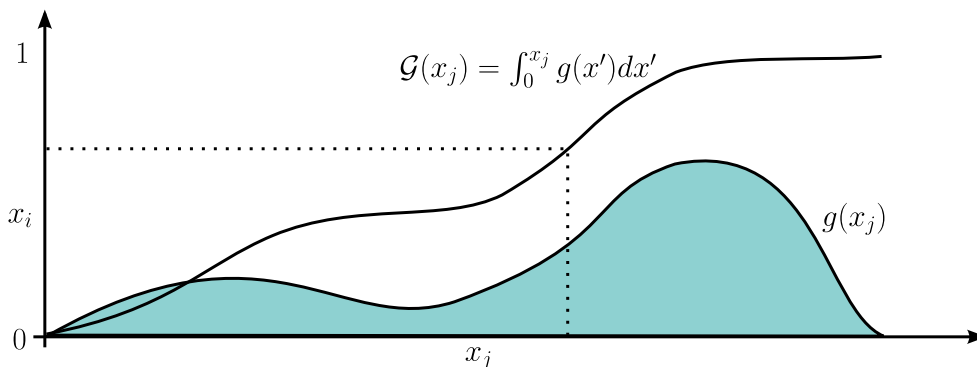
Figure 5: Illustration of the transformation method with the probability density function $g(x)$ and the cumulative distribution function $\mathcal{G}(x)$.

as:

$$\langle A \rangle = \frac{\int_V A(x)p(x)dx}{\int_V p(x)dx} = \frac{\int_V \frac{A(x)p(x)}{g(x)}g(x)dx}{\int_V \frac{p(x)}{g(x)}g(x)dx} \approx \frac{\frac{V}{N}\sum_j^N \frac{A(x_j)p(x_j)}{g(x_j)}}{\frac{V}{N}\sum_j^N \frac{p(x_j)}{g(x_j)}}. \tag{2.14}$$

The random numbers x_j are then chosen according to the probability density function $g(x)$. In other words, we have absorbed the weight $g(x)$ into the random numbers x_j. We need to correct the integrand for this absorption dividing it by $g(x)$. We will now take a look how to generate random numbers x_j according to the probability density $g(x)$. A general approach in distributing random numbers along a probability density function $g(x)$ is called inverse transform sampling [32]. We illustrate this method in Figure 5. For this matter we define the cumulative distribution function $\mathcal{G}(x)$ to our probability density function $g(x)$ with:

$$\mathcal{G}(x) = \int_0^x g(x')dx'. \tag{2.15}$$

A random number x_j with respect to $g(x)$ can now be chosen by $x_j = \mathcal{G}^{-1}(x_i)$, where x_i follows a uniform distribution in the interval $x_i \in [0, 1)$. Remember that a computer usually supplies us with random numbers out of this interval.
If we decide to choose our random numbers x_j from the uniform distribution $g(x) = \frac{1}{V}$, then we find $\mathcal{G}(x) = \frac{x}{V}$, $\mathcal{G}^{-1}(x) = Vx$. As one would expect this results in $x_j = Vx_i$ for the one dimensional case. Furthermore:

$$\langle A \rangle \approx \frac{\frac{1}{N}\sum_j^N A(x_j)p(x_j)}{\frac{1}{N}\sum_j^N p(x_j)}. \tag{2.16}$$

This means we need to find the arithmetic mean of the product $A(x)p(x)$ and $p(x)$ with respect to uniformly distributed random numbers x_j. Setting also $p(x) = $ const recovers equation (2.8). The most general form is given in equation (2.14).

Let us now look at the variance one more time. For a general $p(x)$ and a general $g(x)$ from which we choose the random numbers equation (2.13) becomes [32]:

$$\int_V \frac{A(x)p(x)}{g(x)}g(x)dx = \langle\frac{Ap}{g}\rangle_{MC} \pm \frac{1}{\sqrt{N}}\sigma_{MC}\left(\frac{Ap}{g}\right). \qquad (2.17)$$

If we set $p(x) = 1$ and $g(x) = \frac{1}{V}$, we recover equation (2.13). It turns out that the optimal choice of $g(x)$ is given by:

$$g(x) = \frac{|A(x)p(x)|}{\int_V |A(x)p(x)|dx}. \qquad (2.18)$$

Importance sampling allows us to reduce the variance to a minimum, if we find a probability density function, which follows the functional value $A(x)p(x)$ as closely as possible.

2.1.3 Rejection Sampling, Markov Chains and the Metropolis-Hastings Algorithm

In the last section we have presented a method of choosing random numbers according to a probability density function $g(x)$, called inverse transform sampling. This method does however have some major short-comings. To generate random numbers, we require the inverse of the cumulative distribution function $\mathcal{G}(x)$. Acquiring the inverse of the cumulative distribution function does not necessarily require analytic knowledge of $\mathcal{G}(x)$. Instead we may create a look-up table. If we supply a random number x_i in the interval $x_i \in [0, 1)$, then can find x_k distributed in $g(x)$ by $x_i = \mathcal{G}(x_k)$. Another shortcoming is the cumulative distribution function itself. From its definition $\mathcal{G}(x) = \int_0^x g(x)dx$ we see that its necessary to compute the integral of $g(x)$ up to x, where we require $\int_V g(x) = 1$. In QMC we usually choose the partition function (or something similar) as our probability density function $g(x)$. In this case $g(x)$ is a high dimensional function and is computationally expensive to calculate.

Rejection Sampling
Fortunately, there is another method in order to generate random numbers according to a probability density function $g(x)$, which does not require $\int_V g(x) = 1$. We refer to this method as rejection sampling.
In general the algorithm works as follows: We have a predefined proposal distribution $f(x)$ and an unknown probability distribution $g(x)$ for which we want to sample our random variables. The only condition we require is that $f(x)$ fully encloses $g(x)$, i.e. $g(x) < Mf(x) \quad \forall x$ with $M > 1$. We sample a random point along the proposal distribution $f(x)$ and an additional uniform random variable
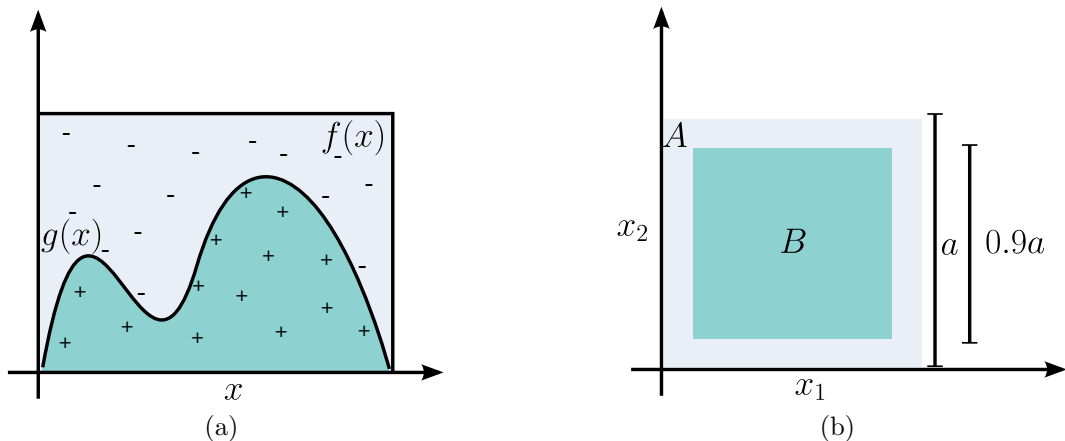
Figure 6: (a) Illustration of rejection sampling for a proposal probability $f(x)$ and probability distribution $g(x)$. (b) Illustration of the dimensionality problem for a hypercube B, representing the acceptance region, enclosed in a hypercube A representing the proposal region.

u. We have generated a random variable x, which follows $g(x)$ if $u \cdot Mf(x) < g(x)$, i.e. the point chosen uniformly along $Mf(x)$ is inside $g(x)$. We illustrate this procedure in Figure 6a. We mark the accepted configurations with a "+" and the rejected ones with a "-". We generate many random variables where $g(x)$ is large and few random variables where $g(x)$ is small.

However, rejection sampling suffers of a dimensionality problem. This can be reasoned when thinking of two hypercubes in d-dimensional space, illustrated in Figure 6b. The hypercube A with side-length a will serve as our proposal region, where we sample uniformly. The enclosed hypercube B with side-length $b = 0.9a$ will serve as our acceptance region. The volume of A scales like $V_A = a^d$, the volume of B scales like $V_B = (0.9a)^d$. For $d \to \infty$ we find $\frac{V_B}{V_A} \to 0$. For very high dimensions, most of the volume of A will be concentrated near its surface. Sampling uniformly in the entire region results in a large amount of rejects.

Markov Chains

From rejection sampling, we take another step towards the Metropolis algorithm. The main ingredient of the Metropolis algorithm is to generate random variables from $g(x)$ using a Markov chain. We do not need to dive too deep into the theory of Markov chains here. A more detailed description can be found in e.g. [33]. The idea of a Markov chain is that we generate random states x' with respect to the current state x (but not any previous states). The fact that a new state is linked to its preceding state in analogy to a chain, inherits the Markov process its name

30
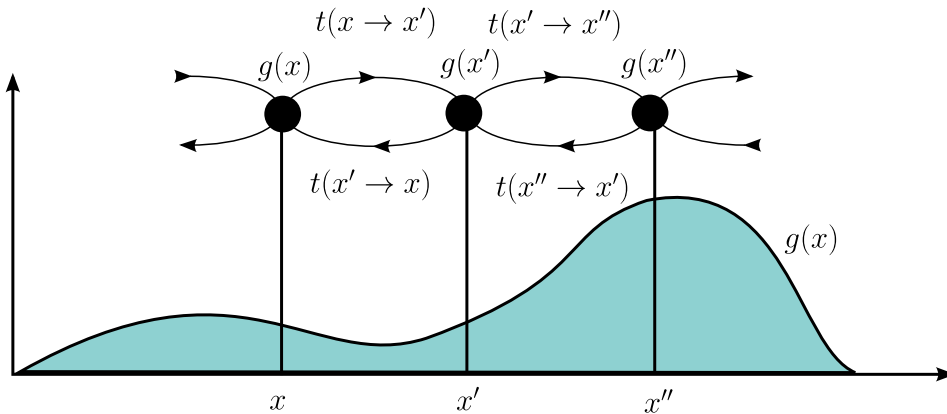
Figure 7: Illustration of a Markov chain to a probability density function $g(x)$. Shown are the states x,x' and x'' and their transition probabilities $t(x \rightarrow x')$, $t(x' \rightarrow x'')$ and the inverse transition probabilities $t(x'' \rightarrow x')$, $t(x' \rightarrow x')$.

as a Markov chain. The transition between a state x and a state x' is defined by the transition probability $t(x \rightarrow x')$. We illustrate this in Figure 7. The Markov chain can be designed, such that it asymptotically reaches $g(x)$ as a stationary distribution, if we guarantee the detailed balance condition and ergodicity.

The detailed balance condition for transition probabilities $t(x \rightarrow x')$ and probability densities $g(x)$ is given by:

$$g(x)t(x \rightarrow x') = g(x')t(x' \rightarrow x). \tag{2.19}$$

One can motivate this condition by thinking of the Markov chain as a random walk through our configuration space $\mathcal{C}$. Then the detailed balance condition assures path-reversibility (i.e. the ability to move the same path in the other direction). With respect to Monte Carlo integration, we can define ergodicity by:

$$\lim_{N \rightarrow \infty} \langle A \rangle_{MC} \rightarrow \langle A \rangle. \tag{2.20}$$

That is, the Monte Carlo expectation value for an infinite amount of steps N in our Markov chain converges to the true expectation value. In a more general definition, a system is ergodic if the mean value of all observables of a system over time (i.e. Markov steps) is the same as the mean value of all observables taken over all possible states. As a result, we find that a Markov chain is ergodic if there exists a non-zero probability to reach any state within a finite number of steps.

Metropolis-Hastings Algorithm
The Metropolis-Hastings algorithm [34, 35] combines the ideas of rejection sampling and Markov chains. Up to this point we have not defined, what the transition

probability $t(x \to x')$ looks like. The main idea of the Metropolis-Hastings algorithm is to split $t(x \to x')$ into an acceptance probability $a(x \to x')$ and a proposal probability $f(x \to x')$:

$$t(x \to x') = a(x \to x')f(x \to x'). \qquad (2.21)$$

We can now insert this expression into the detailed balance equation (2.19). Rearranging for the acceptance rates, we find:

$$\frac{a(x \to x')}{a(x' \to x)} = \frac{g(x')}{g(x)}\frac{f(x' \to x)}{f(x \to x')}. \qquad (2.22)$$

If we are in a state x, we always want to move to the state x' if it is more probable. On the other hand if x' is less probable, we only want to move there according to a probability supplied by the detailed balance equation. Including the ideas of the rejection sampling, we find:

$$a(x \to x') = min\left(1, \frac{g(x')}{g(x)}\frac{f(x' \to x)}{f(x \to x')}\right). \qquad (2.23)$$

This way we generate random numbers x_i according to $g(x)$. While in rejection sampling we found independent random numbers, here a random number x_i depends on the previously found random number. This results in the problem of introducing an auto-correlation length. By this we mean that a configuration in the Markov chain may depend on the previous n configurations, where n is the auto-correlation length. After n Monte Carlo steps a generated configuration becomes independent of the initial configuration. As opposed to rejection sampling, in the Metropolis-Hastings algorithm we, however, do not have the problem of dimensionality.

As a starting point of the Metropolis-Hastings algorithm we choose an arbitrary configuration x_i. It is usually not possible to tell if this configuration x_i is a likely one or an unlikely one (most of the time it will be the latter). In order to reach the likely parts of our configuration space, we need to generate some initial configurations to get there. These initial configurations are not suitable to measure any contribution to the integral, otherwise we would over-estimate unlikely configurations. The number of steps it takes to reach likely configurations are usually called burn-in steps or warm-up steps.

Proposal Probability

We will discuss the proposal probability in more detail here. The proposal probability essentially defines what class of problems we deal with. It defines our configuration space $\mathcal{C}$ to be discrete or continuous. We will apply the Metropolis-Hastings algorithm to three problems in more detail to see how it works.

First, let us integrate some continuous function $f(x_i)$ in $\mathbb{R}^n$, which is only non-zero inside a known region, defined by some vertices v_i in $\mathbb{R}^n$. The probability density function $g(x_i)$, sometimes referred to as the weight, is then just given by a product of θ-functions specifying this region. We can move from one integration point to the next using a random-walk scheme. As soon as a random-walk step is proposed, which leaves the defined region, we reject this step. In this example the configuration space is identical to the specified region.

Next, let us mention the famous example of the Ising model. The weight $g(x)$ is now given by the Boltzmann distribution $e^{-\beta H}$. We propose a walk in our configuration space by selecting a lattice site at random and proposing a spin-flip for this site.

Lastly, we introduce the concept of quantum Monte Carlo (QMC) integration. We have already derived the weight along with the partition function in the previous chapter. For now we will refer to the QMC-weight as $p(k, \tau_1, \ldots, \tau_k)$. The proposal probability of an operator pair insert is then given by the length of our trace β. We will focus on the details of QMC in the following section.

	Probability Density $g(x)$	Proposal Probability $f(x \to x')$
Continuous Function	$\prod_i \theta(x_i - v_i)$	Random Walk: $x_{i'} = r + x_i$
Ising Model	$e^{-\beta H}$	Spin Flip: $\frac{1}{N}$
Quantum Monte Carlo	$p(k, \tau_1, \ldots, \tau_k)$	$dd^\dagger$-Insert: $\frac{d\tau}{\beta}$

2.1.4 Configuration Space $\mathcal{C}$

For now we have discussed Monte Carlo integration with respect to some integration volume V. It is common to refer to the integration volume as the configuration space $\mathcal{C}$. The integrand is usually referred to as the weight w_i. Each weight w_i describes a specific configuration in the configuration space $\mathcal{C}$. The random walk of Markov chains is performed in this configuration space.

2.2 Quantum Monte Carlo Integration

In Chapter 1 and the first part of Chapter 2 we have introduced the basics of the Anderson impurity model and of Monte Carlo integration. Now we combine these two concepts to discuss quantum Monte Carlo (QMC) algorithms. When sampling the trace of the expectation value and the partition function with infinitesimal time intervals $d\tau$, we refer to this as continuous-time Quantum Monte Carlo (CT-QMC). A very detailed summary of CT-QMC algorithms can be found

in [26]. Before CT-QMC algorithms became popular, the Hirsch-Fye algorithm was used to calculate the Anderson Impurity Model [36]. This algorithm requires a discretization of imaginary time. As a results very large grids are necessary for low temperatures. Additionally, there are no generalizations for the interaction to multi-orbital models beyond the density-density interaction due to the restrictions imposed by the Hubbard-Stratonovich transformation [37]. These shortcomings are alleviated in CT-QMC. In the following we will focus on CT-QMC algorithms in the hybridization expansion (1.32) (usually referred to as CT-Hyb), which we derived in the previous chapter.

2.2.1   Partition Function Sampling

First let us rewrite equation (1.32) into a form that is suitable for Monte Carlo integration. We start with the expression we have derived for the partition function:

$$Z = Tr\left[e^{-\beta H_{loc} + H_{bath}}\right] + Z_{bath} \sum_{k\in 2\mathbb{N}} \int_{\tau_{k-1}}^{\beta} d\tau_k \int_{\tau_{k-2}}^{\beta} d\tau_{k-1} \ldots \int_{\tau_1}^{\beta} d\tau_2 \int_0^{\beta} d\tau_1 \times$$

$$Tr_d\left[T_\tau e^{-\beta H_{loc}} d_{\alpha_k}(\tau_k) d_{\alpha_{k-1}}^\dagger(\tau_{k-1}) \ldots d_{\alpha_2}(\tau_2) d_{\alpha_1}^\dagger(\tau_1)\right] \times \det \boldsymbol{\Delta}. \quad (2.24)$$

We split the above expression into a local weight and a weight of the bath:

$$Z = Tr\left[e^{-\beta H_{loc} + H_{bath}}\right] + \sum_{k\in 2\mathbb{N}} \int_{\tau_{k-1}}^{\beta} d\tau_k \int_{\tau_{k-2}}^{\beta} d\tau_{k-1} \ldots \int_{\tau_1}^{\beta} d\tau_2 \int_0^{\beta} d\tau_1 \times$$

$$\underbrace{Tr_d\left[T_\tau e^{-\beta H_{loc}} d_{\alpha_k}(\tau_k) d_{\alpha_{k-1}}^\dagger(\tau_{k-1}) \ldots d_{\alpha_2}(\tau_2) d_{\alpha_1}^\dagger(\tau_1)\right]}_{w_{loc}(k,\tau_1,\ldots,\tau_k)} \times \underbrace{Z_{bath} \det \boldsymbol{\Delta}}_{w_{bath}(k,\tau_1,\ldots,\tau_k)}. \quad (2.25)$$

The entire weight of the partition function in its hybridization expansion (excluding the zeroth order) is found as:

$$p(k,\tau_1,\ldots,\tau_k) = w_{loc}(k,\tau_1,\ldots,\tau_k) w_{bath}(k,\tau_1,\ldots,\tau_k) d\tau_1 \ldots d\tau_k. \quad (2.26)$$

In equation (2.26) we have introduced a relatively severe problem, which is present in all fermionic QMC algorithms. In the derivation of Monte Carlo theory we have determined the probability density function $g(x)$, which serves as the weight of the Metropolis algorithm. In order for a function to serve as a probability density function, we require that $g(x) > 0 \quad \forall x$ and $\int_V g(x) = 1$. The second condition can be neglected in the Metropolis algorithm. The first condition is not fulfilled when $g(x) = p(k,\tau_1,\ldots,\tau_k)$ is the trace of a combination of fermionic

Discrete Trace $\qquad\qquad$ **Continuous Trace**

$$f(k \to k+2) = \frac{\Delta\tau^2}{\beta^2} \qquad\qquad f(k \to k+2) = \frac{d\tau^2}{\beta^2}$$
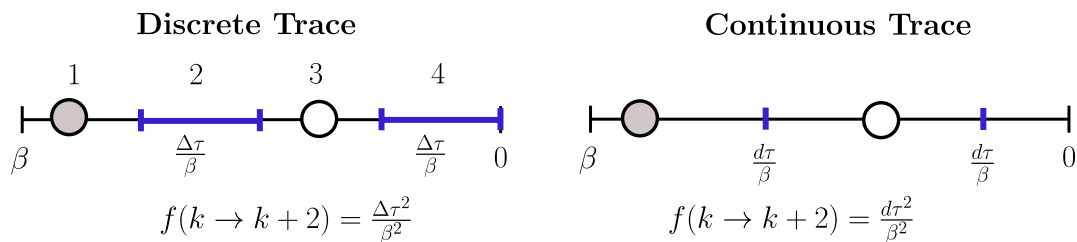
Figure 8: Illustration of the proposal probability for a pair insert. We go from a discrete trace to a continuous trace to motivate the proposal probability $\frac{d\tau}{\beta}$ for every operator as the probability of choosing an infinitesimal interval $d\tau$ from a trace of length β. The grey and white circles resemble existing annihilators and creators in the trace. The blue intervals mark, where the operator pair will be inserted.

operators, as fermionic operators introduce a sign. The common way to deal with this problem is to assume $g(x) = |p(k, \tau_1, \ldots, \tau_k)|$, and then re-weighing our Monte Carlo integration by the mean sign of the trace. This however introduces the so-called sign problem, which in essence is a cancellation problem of the mean sign in the denominator, creating large QMC errors. We will deal with the details of re-weighing and the sign in Section 2.2.3.

Further, notice that the infinitesimal time elements $d\tau_i$ for $i = 1 \ldots k$ are usually considered to be part of the weight $p(k, \tau_1, \ldots, \tau_k)$. This is a convention used in the QMC community. It is very strange to include infinitesimals into the weight, as we considered this to be our probability density function. We will give a short reasoning how the Metropolis condition takes care of this oddity and why this makes sense. In the Metropolis algorithm we only look at ratios of acceptance probabilities. Then all infinitesimals in the weights will cancel out if we propose a move which contains the expansion order $k/2$. If we propose a move which changes the expansion order $k/2 \to k/2+1$ (i.e. adding an operator pair) or $k/2 \to k/2-1$ (i.e. removing an operator pair) the additional infinitesimals $d\tau_i d\tau_j$ in the weight cancel with infinitesimals in the proposal probability. We will look at this in more detail now.

Pair Add and Pair Remove Moves

In order to be ergodic, we need to find moves which allow us to change the expansion order $k/2$. This way we are able to sample the entire configuration space. We remember that the expansion order directly refers to the number of $d, d^\dagger$ pairs in the local trace. The proposal probability for adding a pair of $d_{\alpha_i}, d^\dagger_{\alpha_j}$ operators
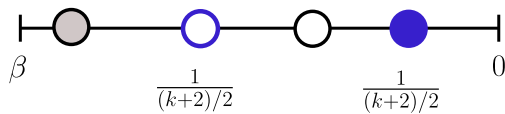
$$f(k+2 \to k) = \frac{1}{((k+2)/2)^2}$$

Figure 9: Illustration of the proposal probability for a pair removal of operators, which were previously added. The number of creation operators and annihilation operators is $\frac{k+2}{2}$ each. The blue operator pair is selected to be removed.

for two infinitesimal time slices of size $d\tau$ (see Figure 8) is:

$$f(k \to k+2) = \frac{d\tau^2}{\beta^2}. \tag{2.27}$$

The weight of a configuration, with an extra operator pair $d_{\alpha_i}, d^\dagger_{\alpha_j}$ is given by:

$$
\begin{aligned}
p(k+2, \tau_1, \ldots, \tau_k; \tau_i, \tau_j) = \\
w_{loc}(k+2, \tau_1, \ldots, \tau_k; \tau_i, \tau_j) w_{bath}(k+2, \tau_1, \ldots, \tau_k; \tau_i, \tau_j) d\tau_1 \ldots d\tau_k d\tau_i d\tau_j. \quad (2.28)
\end{aligned}
$$

We have introduced a semicolon to the notation of the weight to highlight the new operator pair:

$$w(k+2, \tau_1, \ldots, \tau_k; \tau_i, \tau_j) \equiv w(k+2, \tau_1, \ldots, \tau_k, \tau_i, \tau_j). \tag{2.29}$$

The proposal probability for removing the same pair of $d_{\alpha_i}, d^\dagger_{\alpha_j}$ operators (see Figure 9) is:

$$f(k+2 \to k) = \frac{1}{((k+2)/2)^2}. \tag{2.30}$$

The proposal probability for removing a pair may change depending on how random operators are chosen, and we need to be very careful here. The above is valid for a pair of operators in the trace by picking a creation operator from $(k+2)/2$ operators and an annihilation operator from $(k+2)/2$ operators. Different proposal probabilities exist for removing segments in a density-density code.

We point out that our proposal probabilities for inserting and removing operator pairs are per se not constrained by quantum numbers. The constraints due to quantum numbers enter the weight $p(k, \tau_1, \ldots, \tau_k)$. That means, if we insert or remove operator pairs and this results in a forbidden trace, then the trace will simply evaluate to zero and the move will not be accepted.

We have now fully determined all parts of the acceptance ratio in equation (2.22). For going from a specific hybridization configuration of order $k/2$, to one with an

additional pair at τ_i, τ_j we obtain the acceptance ratio:

$$\frac{a(k \rightarrow k+2)}{a(k+2 \rightarrow k)} =$$

$$\frac{|w_{loc}(k+2, \tau_1, \ldots, \tau_k; \tau_i, \tau_j) w_{bath}(k+2, \tau_1, \ldots, \tau_k; \tau_i, \tau_j) d\tau_1 \ldots d\tau_k d\tau_i d\tau_j|}{|w_{loc}(k, \tau_1, \ldots, \tau_k) w_{bath}(k, \tau_1, \ldots, \tau_k) d\tau_1 \ldots d\tau_k|} \times$$

$$\frac{\beta^2}{((k+2)/2)^2 d\tau_i d\tau_j}. \quad (2.31)$$

We observe that all infinitesimals cancel out properly and our solution is still well behaved. We find the Metropolis acceptance rate for a pair insertions hence:

$$a(k \rightarrow k+2) =$$
$$min\left(1, \frac{|w_{loc}(k+2, \tau_1, \ldots, \tau_k; \tau_i, \tau_j) w_{bath}(k+2, \tau_1, \ldots, \tau_k; \tau_i, \tau_j)|}{|w_{loc}(k, \tau_1, \ldots, \tau_k) w_{bath}(k, \tau_1, \ldots, \tau_k)|} \frac{\beta^2}{((k+2)/2)^2}\right).$$
$$(2.32)$$

The Metropolis acceptance rate for a pair removal is then just given by the inverse:

$$a(k+2 \rightarrow k) =$$
$$min\left(1, \frac{|w_{loc}(k, \tau_1, \ldots, \tau_k) w_{bath}(k, \tau_1, \ldots, \tau_k)|}{|w_{loc}(k+2, \tau_1, \ldots, \tau_k; \tau_i, \tau_j) w_{bath}(k+2, \tau_1, \ldots, \tau_k; \tau_i, \tau_j)|} \frac{((k+2)/2)^2}{\beta^2}\right).$$
$$(2.33)$$

We point out that a special matrix update algorithm, called inverse by partitioning, exists for calculating the determinant and inverse of a matrix, such as det $\mathbf{\Delta}$ in w_{bath}, where an additional row and column is added. From a computational viewpoint it is very helpful to employ this method here.

Operator Shift Moves
In principle the insertion and removal updates are sufficient to sample the partition function in an ergodic way. One can imagine that, if there are enough Monte Carlo steps, eventually any allowed configuration along the continuous trace will be sampled. We will nonetheless introduce shift moves as an additional move. Essentially, we pick an operator at random and propose a shift along the trace. In its most general form, the move also allows for the shifted operator to jump past other operators in the trace. This move contains the expansion order with $k/2 \rightarrow k/2$ and may be useful to decrease auto-correlation times. We give a short summary of the acceptance rates as a basis for the worm replacement move in the next chapter, which will be used to tackle a relatively severe ergodicity problem.

$$f'(k \to k) = \frac{1}{(k/2)} \frac{d\tau}{\beta}$$
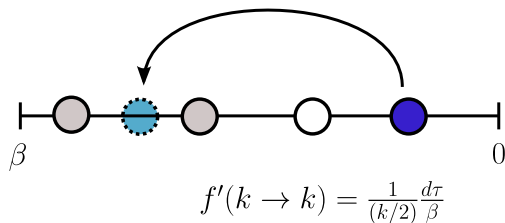
Figure 10: Illustration of the proposal probability for a shift move of a single operator. The grey and white circles resemble existing annihilators and creators in the trace.

The proposal probability for shifting an operator d_{α_i} or $d_{\alpha_i}^\dagger$ is the probability of selecting this specific operator times the probability to put it at a given τ:

$$f'(k \to k) = \frac{1}{(k/2)} \frac{d\tau}{\beta}. \tag{2.34}$$

The proposal probability of shifting the operator back to its original position is exactly the same:

$$f(k \to k) = \frac{1}{(k/2)} \frac{d\tau}{\beta}. \tag{2.35}$$

We illustrate the shift update in Figure 10. Since the proposal probabilities and the infinitesimals $d\tau$ cancel out, the acceptance ratio is fully determined by the ratio of weights. The Metropolis acceptance rate the follows as:

$$a'(k \to k) = min \left(1, \frac{|w_{loc}(k, \tau_1, \ldots, \tau_i', \ldots, \tau_k) w_{bath}(k, \tau_1, \ldots, \tau_i', \ldots, \tau_k)|}{|w_{loc}(k, \tau_1, \ldots, \tau_i, \ldots, \tau_k) w_{bath}(k, \tau_1, \ldots, \tau_i, \ldots, \tau_k)|} \right).$$
$$\tag{2.36}$$

In order to add a row and column to an existing combination of a row and column of a matrix we can employ the Sherman-Morrison formula and the matrix determinant lemma. Again, this is very useful from a computational point of view.

2.2.2 Switchboard Pictures, Segment Pictures and Hybridization Lines

We will now spend some time in showing how configurations of the local trace in CT-Hyb can be visualized. For density-density interactions we can draw segment pictures and switchboard pictures in the occupation number basis (for an illustration see Figure 12). The switchboard picture scales exponentially with $2^{(2 \cdot orbitals)}$ as we have to consider many-particle states. The base 2 is a consequence of the many-particle state being occupied or unoccupied. The exponent 2 stems from the spin degree of freedom. Segment pictures (as in Figure 14), on the other hand, are only build of single-particle states. They not only provide a very elegant way of
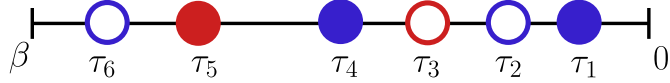
Figure 11: Illustration of a given configuration in the local trace.
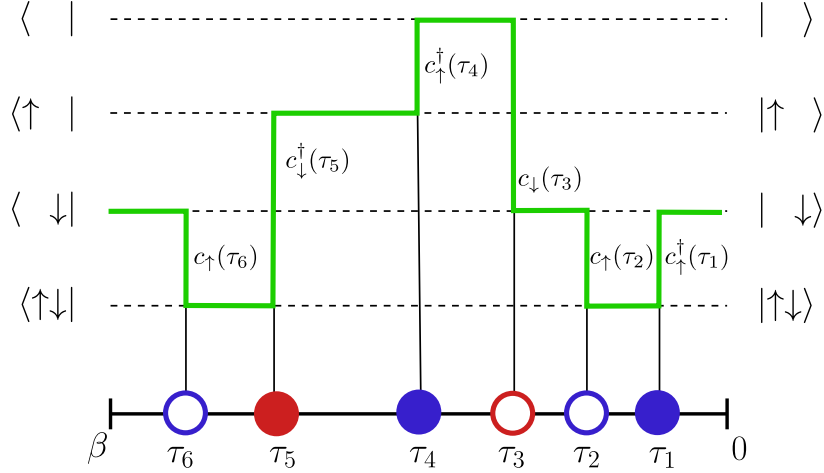
Figure 12: Switchboard Picture for the configuration given in Figure 11.

picturing the trace, but also allow us to introduce some major simplifications to the calculation of the local trace. The algorithm then scales linear with $(2 \cdot \text{Orbitals})$ in the local trace. For general interactions we cannot draw segment pictures as the interaction can rotate the time evolved vector in our trace allowing for cases strictly forbidden in density-density interactions.

Switchboard Pictures

The density-density case results in H_{loc} being diagonal. As a consequence the time evolution $e^{-\tau H_{loc}}$ is easy to calculate. The simplest picture, which we are allowed to draw for any interaction is just one trace as illustrated in Figure 11 for an example configuration.

In this work, we represent creation operators using filled shapes and annihilation operators using empty shapes. We color code spin up operators with blue shapes and spin down operators with red shapes. For the single-orbital case, we restrict ourselves to drawing circles. For more orbitals we will use additional shapes to encode different flavors. We can now follow the path created by the operators through our occupation number states explicitly, as illustrated in Figure 12.

In this case, we observe that the path wraps around a non-empty state $|\downarrow\rangle$. Of course we may also encounter traces, where the path is in the $|\ \rangle$-state at $\tau = 0$ and $\tau = \beta$ as illustrated in Figure 13. This example is however more important for another reason. We notice that a second path exists, which is in the $|\uparrow\rangle$-state at

39
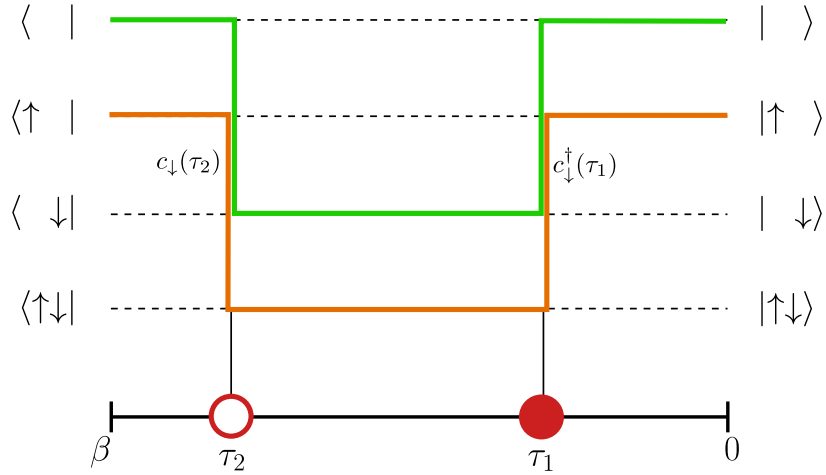
Figure 13: Switchboard Picture for configuration ending in unoccupied state $|\;\;\rangle$ (green path). A second path exists, ending in the singly-occupied state $|\uparrow\rangle$ (orange path).

$\tau = 0$ and $\tau = \beta$. We can now make the following observation: in the switchboard picture we are required to draw all possible paths, which contribute to the trace.

Segment Pictures

We are now able to take a step from switchboard pictures to segment pictures, as illustrated in Figure 14. We will introduce some terminology with reference to Figure 14. We usually refer to the occupied state between the two operators $d_\uparrow(\tau_2)d_\uparrow^\dagger(\tau_1)$ with $\tau_2 > \tau_1$ as a segment. In this example an anti-segment is given by the unoccupied state between the two operators $d_\downarrow^\dagger(\tau_5)d_\downarrow(\tau_3)$, where $\tau_5 > \tau_3$. Further notice, that we are not drawing the states $|\;\;\rangle$ and $|\uparrow\downarrow\rangle$ anymore. Instead, the state $|\;\;\rangle$ is given if both $|\uparrow\rangle$ and $|\downarrow\rangle$ are empty and the state $|\uparrow\downarrow\rangle$ is given by the overlaps.

Let us reconsider the density-density type local Hamiltonian introduced in Chapter 1 for the single-orbital case:

$$H_{loc} = -\mu\left(n_\uparrow + n_\downarrow\right) + U n_\uparrow n_\downarrow. \tag{2.37}$$

We notice that the term $-\mu$ only contributes while a state is occupied. This directly translates to the total length of the segments $l_\uparrow$ and $l_\downarrow$. We further notice that the term U only contributes while both state $|\uparrow\rangle$ and $|\downarrow\rangle$ are occupied. This directly translates to the overlap of the segments $s_{\uparrow\downarrow}$. For the local trace we find:

$$Tr_d\left[e^{-\beta H_{loc}}d_{\alpha_k}(\tau_k)d_{\alpha_k}^\dagger(\tau_k')\ldots d_{\alpha_1}(\tau_1)d_{\alpha_1}^\dagger(\tau_1')\right] = e^{-\beta\left(-\mu\left(l_\uparrow + l_\downarrow\right) + U s_{\uparrow\downarrow}\right)}. \tag{2.38}$$

40
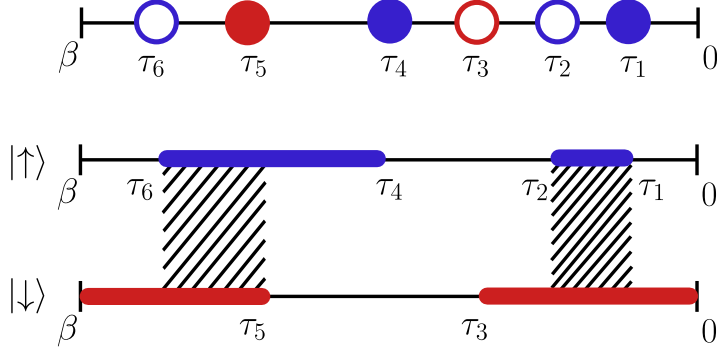
Figure 14: Segment Picture for the configuration given in Figure 11. For $|\uparrow\rangle$ there are two segments, for $|\downarrow\rangle$ there is an anti-segment. The doubly occupied state $|\uparrow\downarrow\rangle$ is symbolized by the overlap regions shaded.

In this work we are interested in general interactions and we will hence not go into the details of CT-Hyb segment codes. We introduced segment pictures as a simple visualization tool. The density-density case gives us the chance to analyze the local trace relatively easy. Especially with regards to the fermionic sign, the density-density case is very valuable as the sign of each configuration is positive here [38]. We will deal with the fermionic sign in the next section.

Hybridization lines
We will now show how the hybridization part of the expansion can be included into pictures of the local trace. To this end, we connect all creation operators with all annihilation operators. Each hybridization line refers to an entry in the hybridization matrix. We will illustrate how hybridization lines connect with our example trace.
The general form of the hybridization matrix for the example trace in Figure 15 has the form:

$$\mathbf{\Delta} = \begin{pmatrix} \Delta_{11}(\tau_1 - \tau_2) & \Delta_{12}(\tau_1 - \tau_3) & \Delta_{13}(\tau_1 - \tau_6) \\ \Delta_{21}(\tau_4 - \tau_2) & \Delta_{22}(\tau_4 - \tau_3) & \Delta_{23}(\tau_4 - \tau_6) \\ \Delta_{31}(\tau_5 - \tau_2) & \Delta_{32}(\tau_5 - \tau_3) & \Delta_{33}(\tau_5 - \tau_6) \end{pmatrix}, \tag{2.39}$$

where we have assigned the times $\tau_1, \ldots \tau_6$ from right to left. Sometimes we will assume that our hybridization function Δ_{ij} is diagonal in orbital and spin (Figure 16). We illustrate how the hybridization lines would then look for our example:
The hybridization matrix then simplifies to:

$$\mathbf{\Delta} = \begin{pmatrix} \Delta_{11}(\tau_1 - \tau_2) & 0 & \Delta_{13}(\tau_1 - \tau_6) \\ \Delta_{21}(\tau_4 - \tau_2) & 0 & \Delta_{23}(\tau_4 - \tau_6) \\ 0 & \Delta_{32}(\tau_5 - \tau_3) & 0 \end{pmatrix}. \tag{2.40}$$

41
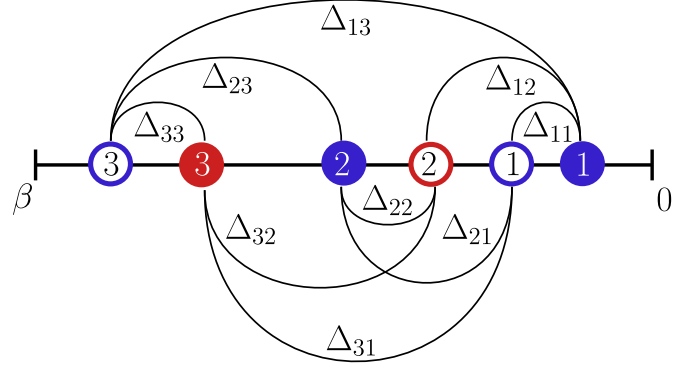
Figure 15: Local trace with all possible hybridization lines connecting annihilation $d_{\alpha_i}(\tau_m)$ and creation operators $d^\dagger_{\alpha_i}(\tau_n)$. We number all creation operators from 1 to $k/2$ and all annihilation operators from 1 to $k/2$. Hybridization events, where $\tau_m > \tau_n$ are drawn above the trace, and hybridization events where $\tau_m < \tau_n$ are drawn beneath the trace.
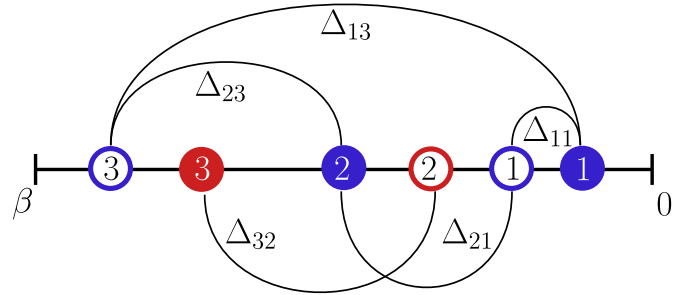
Figure 16: Local trace with hybridization events of a diagonal hybridization function $\Delta_{\sigma_i \sigma_j}$.

When measuring the one-particle Green's function two operators are local operators appearing in the trace. For the $2n$-point Green's function $2n$ operators appear locally. In partition function space $\mathcal{C}_Z$ we can generate such local operators by cutting hybridization lines and then measure the resulting weight.

2.2.3 The Sign of Sampling in Quantum Monte Carlo

As mentioned before, fermionic QMC calculations include an additional difficulty during sampling due to the sign of the weight $p(k, \tau_1, \ldots, \tau_k)$. The root of the sign problem is found in the fermionic anti-commutation rule. When exchanging the order of any two operators (at different times) we produce an additional sign. In the previous section we have expanded the partition function and the closely related thermal expectation value. In order to sample the expectation value we use importance sampling around the weight $p(k, \tau_1, \ldots, \tau_k)$. Interpreting these weights as probabilities for our Metropolis algorithm requires them to be positive, which is not necessarily the case. Our starting point is the general equation (2.16):

$$\langle A \rangle \approx \frac{\frac{1}{N} \sum_j^N A(x_j) p(x_j)}{\frac{1}{N} \sum_j^N p(x_j)}, \tag{2.41}$$

where the random numbers x_j are chosen uniformly over our whole integration volume. We now set $p(x_j) = \text{sign}(p(k, \tau_1, \ldots, \tau_k))|p(k, \tau_1, \ldots, \tau_k)|$ such that:

$$\langle A \rangle \approx \frac{\frac{1}{N} \sum A(k, \tau_1, \ldots, \tau_k) \text{sign}(p(k, \tau_1, \ldots, \tau_k))|p(k, \tau_1, \ldots, \tau_k)|}{\frac{1}{N} \sum_j^N \text{sign}(p(k, \tau_1, \ldots, \tau_k))|p(k, \tau_1, \ldots, \tau_k)|}. \tag{2.42}$$

Now we can employ importance sampling by choosing random numbers x_l according to $|p(k, \tau_1, \ldots, \tau_k)|$, using e.g. the Metropolis algorithm:

$$\langle A \rangle \approx \frac{\frac{1}{N} \sum_l^N A(x_l) \text{sign}(p(x_l))}{\frac{1}{N} \sum_l^N \text{sign}(p(x_l))}. \tag{2.43}$$

We find that we need to re-weigh the measurement by the average sign $\langle \text{sign} \rangle$. In this section we attempt to give a deeper insight into how the sign of the current configuration is calculated. This usually is a non-trivial task and needs to be done with great care. For the Metropolis algorithm we have found that:

$$p(k, \tau_1, \ldots, \tau_k) = w_{loc}(k, \tau_1, \ldots, \tau_k) w_{bath}(k, \tau_1, \ldots, \tau_k) d\tau_1 \ldots d\tau_k. \tag{2.44}$$

We observe that the sign can come from several contributions, which can be divided into contributions from the local trace and contributions from the hybridization part. We will first focus on the signs evolving from the local trace and then on the signs evolving from the hybridization part.

Sign of the local time ordered Trace s_{loc}

When calculating the local time ordered trace in a straight forward way we may encounter an overall sign. We will show this for our sample trace in Figure 11. When applying the time ordering operator the local trace is given by:

$$w_{loc} = Tr\left[e^{-\beta H_{loc}}d_\uparrow(\tau_6)d_\downarrow^\dagger(\tau_5)d_\uparrow^\dagger(\tau_4)d_\downarrow(\tau_3)d_\uparrow(\tau_2)d_\uparrow^\dagger(\tau_1)\right]. \qquad (2.45)$$

We now rewrite the time dependent operators in the interaction picture explicitly according to (1.11), where our time evolution now only covers the local Hamiltonian H_{loc}.

$$w_{loc} = Tr\left[e^{-(\beta-\tau_6)H_{loc}}d_\uparrow e^{-(\tau_6-\tau_5)H_{loc}}d_\downarrow^\dagger e^{-(\tau_5-\tau_4)H_{loc}}d_\uparrow^\dagger\times\right.$$

$$\left. e^{-(\tau_4-\tau_3)H_{loc}}d_\downarrow e^{-(\tau_3-\tau_2)H_{loc}}d_\uparrow e^{-(\tau_2-\tau_1)H_{loc}}d_\uparrow^\dagger e^{-\tau_1 H_{loc}}\right]. \qquad (2.46)$$

When writing creation and annihilation operators in the occupation number basis these operators correspond to sparse matrices. The entries need not be all positive! We will show this for our 4 dimensional Hilbert space of two flavors, i.e. spin-up and spin-down. We assume the orthonormal basis:

$$|\ \rangle = \begin{pmatrix}1\\0\\0\\0\end{pmatrix} \quad |\uparrow\ \rangle = \begin{pmatrix}0\\1\\0\\0\end{pmatrix} \quad |\ \downarrow\rangle = \begin{pmatrix}0\\0\\1\\0\end{pmatrix} \quad |\uparrow\downarrow\rangle = \begin{pmatrix}0\\0\\0\\1\end{pmatrix}. \qquad (2.47)$$

We need to choose a sign convention to fulfill the fermionic anti-commutation relations $\{d_i, d_j^\dagger\} = \delta_{ij}$, $\{d_i, d_j\} = 0$ and $\{d_i^\dagger, d_j^\dagger\} = 0$. One way to go about this, is to find the correct representation of the annihilation operators first. We add a sign to d_i with $(-1)^N$, where N is the number of states $k < i$ being occupied. In case of having only two spins $\{\uparrow, \downarrow\}$ this adds a sign for $d_\downarrow |\uparrow\downarrow\rangle = -|\uparrow\ \rangle$. Once all annihilation operators are set up, we find the creation operators by building the conjugate transpose of the annihilation operators.

With this convention we find:

$$d_\uparrow = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad d_\downarrow = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \qquad (2.48)$$

$$d_\uparrow^\dagger = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \quad d_\downarrow^\dagger = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{pmatrix}. \qquad (2.49)$$

If we were to ignore the time evolution in between the matrices, then the product of our six matrices in the trace would result in a negative sign in one component:

$$d_\uparrow \times d_\downarrow^\dagger \times d_\uparrow^\dagger \times d_\downarrow \times d_\uparrow \times d_\uparrow^\dagger = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}. \tag{2.50}$$

Let us now focus on why we are allowed to ignore the time evolution in between the matrices in this case. The time evolution of operators in the local trace is given by $e^{-\tau_i H_{loc}}$, where $\tau_i > 0$. In order to calculate the matrix exponential for general interactions U, it is convenient to diagonalize H_{loc}. The matrix exponential for a diagonalizable matrix A and a diagonal matrix of its eigenvalues D with $A = T^{-1}DT$ is given by:

$$e^{sA} = \sum_n^\infty \frac{(sA)^n}{n!} = sA + \frac{(sA)^2}{2} + \frac{(sA)^3}{6} + \dots$$

$$= sT^{-1}DT + \frac{s^2}{2}T^{-1}D\underbrace{TT^{-1}}_{1}DT + \frac{s^3}{6}T^{-1}D\underbrace{TT^{-1}}_{1}D\underbrace{TT^{-1}}_{1}DT + \dots$$

$$= T^{-1}\left(\sum_n^\infty \frac{(sD)^n}{n!}\right)T = T^{-1}e^{sD}T, \tag{2.51}$$

where s is a scalar. The matrix exponential of a diagonal matrix is the exponential of each diagonal entry. We usually subtract the lowest eigenvalue in the exponent to shift our problem relative to the zero-point energy. This is done for the exponential to stay well-behaved for $\beta \to \infty$, as otherwise the partition function $Z = \text{Tr}e^{-\beta H} = \sum_i e^{-\beta \varepsilon_i}$ diverges.

In any case the time evolution, defined by a matrix exponential, will always stay positive definite. The sign in the local trace is just a result of the multiplication done in equation (2.50).

Sign of ordering of Trace into alternating Pairs s_{alt}

When not considering the trace in its time ordered form, but the form derived in the hybridization expansion in formula (1.32), then we are required to write creation and annihilation operators in alternating order for each flavor:

$$Z = Z_{bath}\sum_{k=0}^\infty \int_{\tau_{k-1}}^\beta d\tau_k \int_{\tau_{k-2}}^\beta d\tau_{k-1} \dots \int_{\tau_1}^\beta d\tau_2 \int_0^\beta d\tau_1 \times$$

$$Tr_d\left[T_\tau e^{-\beta H_{loc}}d_{\alpha_k}(\tau_k)d_{\alpha_{k-1}}^\dagger(\tau_{k-1})\dots d_{\alpha_2}(\tau_2)d_{\alpha_1}^\dagger(\tau_1)\right] \times \det\mathbf{\Delta}. \tag{2.52}$$

This is indeed very important and is a result of the hybridization matrix requiring an alternating order when connecting creation to annihilation operators. This can be seen in the following example, where we assume two local traces of the form:

$$Tr_{d_1}[d_{\alpha_1}(\tau_4)d_{\alpha_1}(\tau_3)d^\dagger_{\alpha_1}(\tau_2)d^\dagger_{\alpha_1}(\tau_1)] \tag{2.53}$$

$$Tr_{d_2}[d_{\alpha_1}(\tau_4)d^\dagger_{\alpha_1}(\tau_2)d_{\alpha_1}(\tau_3)d^\dagger_{\alpha_1}(\tau_1)]. \tag{2.54}$$

We number the creation operators $d^\dagger_{\alpha_1}(\tau_1), d^\dagger_{\alpha_1}(\tau_2)$ with $1, 2$ and the annihilation operators $d_{\alpha_1}(\tau_3), d_{\alpha_1}(\tau_4)$ with $1, 2$. The hybridization matrix connects creation to annihilation operators. We label the rows according to the number of creation operators and the columns according to the number of annihilation operators. Both traces Tr_{d_1} and Tr_{d_2} give a hybridization matrix of the form:

$$\mathbf{\Delta} = \begin{pmatrix} \Delta_{11}(\tau_1 - \tau_3) & \Delta_{12}(\tau_1 - \tau_4) \\ \Delta_{21}(\tau_2 - \tau_3) & \Delta_{22}(\tau_2 - \tau_4) \end{pmatrix}, \tag{2.55}$$

where for now we ignore any possible quantum number violations. In fact in the density-density case the correct alternating order would be enforced by the quantum number constraints and equation (2.53) would not be possible. For the density-density case, we just need to check if each state is filled by segments or by anti-segments and then add an overall sign for one of the two cases. In the case of general interactions, the ordering becomes much more important.

Trace Tr_{d_1} and Tr_{d_2} are equivalent up to a position change of the two operators at τ_2 and τ_3 (i.e. before the time ordering sign is applied). This results in a fermionic sign change due to the anti-commutator. This sign change cannot be recovered by the hybridization matrix. We can fix this problem by requiring an alternating order for the operator pairs of the same flavor.

We will now take another look at our example trace in Figure 11. We can observe that the alternating order $d_\uparrow d^\dagger_\uparrow$ is already fulfilled for the spin-up flavor. The spin down flavor is not in the required $d_\downarrow d^\dagger_\downarrow$ order and the operators need to be flipped once, resulting in an overall sign.

Sign of Hybridization Matrix Determinant $s_{\mathbf{det}}$
The determinant of the hybridization matrix itself can and typically will also produce a sign, since the hybridization function Δ is anti-periodic.

Sign of sorting Hybridization Matrix into block structure $s_{\mathbf{block}}$
The following signs result from the technical implementations of the algorithm. If we are considering a hybridization, which is diagonal in flavor, then we are able to build a block diagonal hybridization matrix. In order to exploit the advantages of calculating the determinant of this block diagonal matrix by just calculating the

product of the determinant of its sub-blocks, we need to re-sort the hybridization matrix connected to the time ordered local trace into this form. We pick up a sign for the exchange of any two operators due to the anti-commutator of fermions. For our example, this would mean that we sort the hybridization matrix of equation (2.40) into a block diagonal form by exchanging column 2 and 3:

$$\mathbf{\Delta} = \begin{pmatrix} \Delta_{11}(\tau_1 - \tau_2) & \Delta_{13}(\tau_1 - \tau_6) & 0 \\ \Delta_{21}(\tau_4 - \tau_2) & \Delta_{23}(\tau_4 - \tau_6) & 0 \\ 0 & 0 & \Delta_{32}(\tau_5 - \tau_3) \end{pmatrix}. \tag{2.56}$$

Again this gives us an overall sign.

Sign of Inverse by Partitioning $s_{\mathbf{inv}}$
Another rather technical sign results when calculating the determinant ratio using inverse by partitioning. For this purpose we sort the newly inserted operators or the ones to be removed to the back of our determinant block. We pick up a sign for the exchange of any two operators due to the anti-commutator of fermions. In order to extract the sign of the current determinant from the determinant ratio, we need to store the sign of the previous determinant. We cannot directly recover the correct sign of the determinant from inverse by partitioning.

Total Sign
We are now able to measure the overall sign of our configuration as a combination of all signs we have discussed:

$$s_{tot} = s_{loc} s_{alt} s_{det} s_{block}. \tag{2.57}$$

We point out that s_{loc}, s_{alt} and s_{det} are signs which are encountered in any CT-Hyb code. The sign s_{block} is implementation dependent and is generated when exploiting computationally feasible algorithms. It turns out that the sign is consistent for all configurations for the single-impurity Anderson impurity model for density-density type interactions [38]. Empirically it was found for CT-Hyb that the sign is also very close to being positive for all configurations for the Kondo lattice and two-orbital models in Reference [27].

3 Worm Sampling

We will now turn to the main topic of this work. In the last chapter we have discussed the QMC algorithm in its hybridization expansion. Instead of using importance sampling in the partition function space $\mathcal{C}_Z$, we will now sample in the Green's function space $\mathcal{C}_{G^{(n)}}$. In principle, worm sampling allows us to insert any combination of operators into the local trace and we are not necessarily limited to Green's function operators. In that sense it is a much more general concept than what will be explained here. Worm sampling algorithms play a minor role in the QMC community. Early worm algorithms for Hirsch-Fye can be found in [39] and for CT-QMC in [40]. Newer implementations can be found in [26] and especially in the CT-Bold algorithm in [41]. To our knowledge the worm algorithm has not yet been implemented to measure two-particle quantities.

3.1 Motivation

Before going into any theoretical details we will give a short motivation why we attempt to measure Green's functions $G^{(n)}$ with worm sampling. When measuring the Green's function $G^{(1)}(\tau)$ in partition function space $\mathcal{C}_Z$ the estimator takes on the form:

$$G_{\mathcal{C}_Z}^{(1)}(\tau) = \langle \sum_{ij} \frac{\det \mathbf{\Delta}^{(ij)}}{\det \mathbf{\Delta}} \text{sgn} \cdot \delta(\tau, \tau_i - \tau_j) \rangle_{MC}, \tag{3.1}$$

where $\Delta^{(ij)}$ is the hybridization matrix with the i-th row and j-th column removed. This estimation by means of a reduced hybridization matrix reveals a first shortcoming of the estimator: the estimator in equation (3.1) fails for strong insulating cases, since in this case only very few operators are found in the trace and few hybridization lines are present. We will see that the estimator of worm sampling does not depend on the determinant ratio of the hybridization matrix. This suggests that the estimator of worm sampling does a better job in sampling strong insulating cases. We would like to make the reader aware of one method to improve the estimator in partition function sampling, sometimes referred to as remove-shift measurement (or sliding measurement), which has been implemented for density-density codes [42]. In this approach the creation and annihilation operators with no hybridization lines attached are shifted inside the trace. As a result, each Green's function contribution includes more information than a δ-function measurement does.

When viewing the estimator in equation (3.1) we further observe that we are restricted to diagrams produced by partition function sampling. For diagonal hybridization functions we are not able to calculate off-diagonal Green's function contributions with this estimator. While off-diagonal contributions do not exist
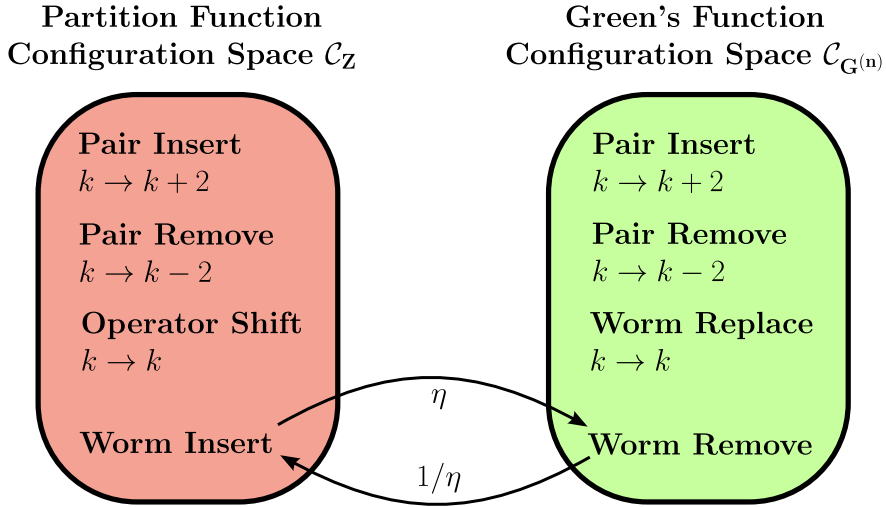
Figure 17: Illustration of the Monte Carlo moves in each configuration space $\mathcal{C}_Z$ and $\mathcal{C}_{G^{(n)}}$ and the transition moves "worm insert" and "worm remove".

for the one-particle Green's function $G^{(1)}$ they exist for the two-particle Green's function $G^{(2)}$, when assuming interactions that are not only of density-density type. When extracting the self-energy $\Sigma(i\omega)$ not by the Dyson equation, but through the equation of motion, a better high-frequency behavior was observed. This method is usually referred to as improved estimators and has so far only been implemented for density-density interactions [43]. Off-diagonal contributions of the Green's function become especially important when implementing improved estimators for two-particle quantities. A long-term goal would be to extract the fully irreducible vertex Λ from the QMC data with the correct high-frequency behavior using these new improved estimators. This is needed for methods such as DΓA [10], requiring Λ properly calculated with QMC algorithms.

3.2 Ergodicity in $\mathcal{C}_Z$ and $\mathcal{C}_{G^{(n)}}$

As already mentioned, in this work we will restrict ourselves to sampling the one-particle Green's function $G^{(1)}_{\alpha_1\alpha_2}(\tau_1, \tau_2)$ and the two-particle Green's function $G^{(2)}_{\alpha_1\alpha_2\alpha_3\alpha_4}(\tau_1, \tau_2, \tau_3, \tau_4)$, as illustrated in Figure 17.
Restricting our Worm sampling to the Green's functions space $\mathcal{C}_{G^{(n)}}$ has two reasons: (1) the one and two-particle Green's functions include almost all relevant information about the quantum impurity. Knowing these two functions allows us to calculate self-energies, susceptibilities, etc. (2) when sampling the one- and two-particle Green's function, we can easily check our results against the measurements in the partition function space $\mathcal{C}_Z$ (especially with regards to the normalization).

Comparing the two sampling methods in $\mathcal{C}_Z$ and $\mathcal{C}_{G^{(n)}}$ then allows us to assess the advantages and disadvantages of each method.

In Figure 17 we have illustrated the configuration space of the partition function $\mathcal{C}_Z$ and the configuration space of a general Green's function $\mathcal{C}_{G^{(n)}}$. We included all steps needed to be ergodic and to decrease auto-correlation lengths in both configuration steps. The pair insertion and removal steps in $\mathcal{C}_Z$ were already discussed in the previous chapter. There we have also introduced the operator shift move for $\mathcal{C}_Z$. We will now give a short summary of all other steps mentioned in the illustration. We can set up a modified partition function $Z_{G^{(n)}}$ in the configurations space $\mathcal{C}_{G^{(n)}}$ by integrating over all degrees of freedom of the Green's function $G^{(n)}$ [26]:

$$ Z_{G^{(n)}} := \oint G^{(n)}_{\alpha_1,\dots,\alpha_n}(\tau_1,\dots\tau_n) = \sum_{\alpha_1,\dots,\alpha_n} \int_{\tau_1} \dots \int_{\tau_n} G^{(n)}_{\alpha_1,\dots,\alpha_n}(\tau_1,\dots\tau_n). \qquad (3.2) $$

We stress that this is not a "physical" partition function in the sense that it has the same meaning as Z. We define $Z_{G^{(n)}}$ to make the two configuration spaces $\mathcal{C}_Z$ and $\mathcal{C}_{G^{(n)}}$ comparable.

3.2.1 Worm Insertion and Removal Steps

The worm insertion and removal steps are transition steps between the two configuration steps. In order to sample in $\mathcal{C}_Z$ and $\mathcal{C}_{G^{(n)}}$ we need to switch between these configuration spaces. We will give a more detailed discussion why it is necessary to sample in both spaces in the next section. We further assume that the configuration spaces $\mathcal{C}_Z$ and $\mathcal{C}_{G^{(n)}}$ may have very different values for their probability density function. We balance these difference by introducing a weighting factor η:

$$ W = Z + \eta Z_{G^{(n)}}. \qquad (3.3) $$

Then our combined sampling space takes on the form:

$$ \mathcal{C} = \mathcal{C}_Z \cup \mathcal{C}_{G^{(n)}}. \qquad (3.4) $$

For now we have not formalized how η scales with the number of orbitals, temperature and interaction strength. It is best to choose η, such that we spend and equal amount of steps in $\mathcal{C}_Z$ and $\mathcal{C}_{G^{(1)}}$. We will revisit this fact when discussing the normalization of the worm result in the following section. We illustrate a worm pair of two operators in a local trace in the Figure 18.

We would like to point out that the only difference between worm operators and existing operators is the missing of hybridization lines. This has some implications

Figure 18: Illustration of a worm pair marked in green inside a local trace. Operators in grey represent operators with hybridization lines connected.

for our Metropolis acceptance rates. The proposal rate of inserting a worm is given by the same expression as the proposal rate of inserting a regular operator pair:

$$f(\mathcal{C}_Z \rightarrow \mathcal{C}_{G^{(1)}}) = \frac{d\tau^2}{\beta^2}. \tag{3.5}$$

Adding a worm pair results in the expansion order $k/2$ of the local trace being increased by 1, whereas the expansion order in the determinant is kept constant. This adds an ambiguity to the expansion order, which we need to keep in mind. The weight of a configuration in $\mathcal{C}_{G^{(n)}}$ modified by η is then:

$$p(\mathcal{C}_{G^{(1)}}, \tau_1, \ldots, \tau_k; \tau_i, \tau_j) =$$
$$\eta \cdot w_{loc}(k+2, \tau_1, \ldots, \tau_k; \tau_i, \tau_j) w_{bath}(k, \tau_1, \ldots, \tau_k) d\tau_1 \ldots d\tau_k. \tag{3.6}$$

We point out that combining the proposal probability and the configuration of the weight, the two infinitesimals $d\tau$ do not cancel as they would have in partition function sampling. This is an artifact of how measurement and sampling are split in partition function sampling and how they are split in worm sampling. In fact, we encounter a similar problem when measuring the Green's function $G^{(1)}$ in partition function space. We identify the two infinitesimals $d\tau$ with belonging to the binning procedure during measurement in τ or belonging to the integration variables during the measurement in $i\omega$ when Fourier transforming. The proposal probability for removing the worm is simply:

$$f(\mathcal{C}_{G^{(1)}} \rightarrow \mathcal{C}_Z) = 1. \tag{3.7}$$

Note, since there is only one worm in the trace at the same time, we will always propose to remove exactly this worm. For the Metropolis acceptance rate we hence find:

$$a(\mathcal{C}_Z \rightarrow \mathcal{C}_{G^{(1)}}) = min\left(1, \eta \cdot \frac{|w_{loc}(k+2, \tau_1, \ldots, \tau_k; \tau_i, \tau_j)|}{|w_{loc}(k, \tau_1, \ldots, \tau_k)|} \cdot \beta^2\right). \tag{3.8}$$

We observe that the bath weight w_{bath}, which includes the hybridization matrix, cancels out due to the fact that the bath remains unchanged.

51

The inverse gives the acceptance probability of a worm removal:

$$a(\mathcal{C}_{G^{(1)}} \to \mathcal{C}_Z) = min\left(1, \frac{1}{\eta} \cdot \frac{|w_{loc}(k, \tau_1, \ldots, \tau_k)|}{|w_{loc}(k+2, \tau_1, \ldots, \tau_k; \tau_i, \tau_j)|} \cdot \frac{1}{\beta^2}\right). \tag{3.9}$$

The same is true for adding two worm operator pairs to the trace, which would result in sampling two-particle quantities. The acceptance rate for the worm insertion move are then given by:

$$a(\mathcal{C}_Z \to \mathcal{C}_{G^{(2)}}) = min\left(1, \eta \cdot \frac{|w_{loc}(k+4, \tau_1, \ldots, \tau_k; \tau_p, \tau_q, \tau_r, \tau_s)|}{|w_{loc}(k, \tau_1, \ldots, \tau_k)|} \cdot \beta^4\right). \tag{3.10}$$

The acceptance rate for the worm removal move is given by:

$$a(\mathcal{C}_{G^{(2)}} \to \mathcal{C}_Z) = min\left(1, \frac{1}{\eta} \cdot \frac{|w_{loc}(k, \tau_1, \ldots, \tau_k)|}{|w_{loc}(k+4, \tau_1, \ldots, \tau_k; \tau_p, \tau_q, \tau_r, \tau_s)|} \cdot \frac{1}{\beta^4}\right). \tag{3.11}$$

In principle, we are allowed to absorb the factor $1/\beta^{2n}$ for the Green's function $G^{(n)}$ into the weighting factor η.

3.2.2 Pair Insertion and Removal Steps in $\mathcal{C}_{G^{(n)}}$

In order to determine the likelihood of a worm configuration we need to do additional sampling in the Green's function space $\mathcal{C}_{G^{(n)}}$. This is a very crucial part of worm sampling. If we forget to sample in Green's function space then we simply jump back and forth between $\mathcal{C}_Z$ and $\mathcal{C}_{G^{(n)}}$, we increase our auto-correlation length to a point, where we start to violate ergodicity. We remind the reader of the expectation value:

$$\langle \mathcal{O}(\tau) \rangle = \frac{1}{Z} Tr\left(T_\tau e^{-\beta H} \mathcal{O}(\tau)\right). \tag{3.12}$$

When sampling in the space of the operator $\mathcal{O}$, we effectively sample the partition function with the additional local operator $\mathcal{O}$ in the trace. This explains why we are required to sample the Green's function space $\mathcal{C}_{G^{(n)}}$ with operators having hybridization lines attached. Acceptance rates look identical to acceptance rates in $\mathcal{C}_Z$ space:

$$a(\mathcal{C}_{G^{(1)}}; k \to k+2) = min\left(1, \frac{|w_{loc}(k+4, \tau_1, \ldots, \tau_m, \ldots, \tau_n, \ldots \tau_k; \tau_i, \tau_j)|}{|w_{loc}(k, \tau_1, \ldots, \tau_m, \ldots, \tau_n, \ldots \tau_k)|} \times \right.$$
$$\left. \frac{|w_{bath}(k+2, \tau_1, \ldots, \tau_k; \tau_i, \tau_j)|}{|w_{bath}(k, \tau_1, \ldots, \tau_k)|} \frac{\beta^2}{((k+2)/2)^2}\right), \tag{3.13}$$
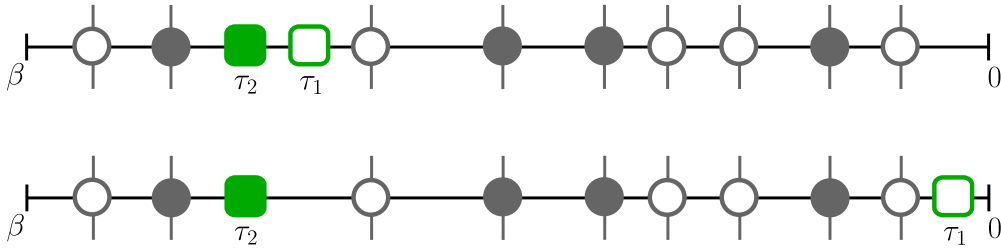
Figure 19: Illustration of the ergodicity problem of a sample trace with many operators. Inserting a worm into the upper trace is relatively easy. Inserting a worm into the lower trace may be forbidden due to quantum number violations.

where the worm operators are at times τ_m, τ_n. The Metropolis acceptance rate for a pair removal in the Green's function space is then just given by the inverse:

$$a(\mathcal{C}_{G^{(1)}}; k + 2 \rightarrow k) = min\left(1, \frac{|w_{loc}(k, \tau_1, \ldots \tau_m, \ldots, \tau_n, \ldots \tau_k)|}{|w_{loc}(k + 4, \tau_1, \ldots \tau_m, \ldots, \tau_n, \ldots \tau_k; \tau_i, \tau_j)|} \times \right.$$
$$\left. \frac{|w_{bath}(k, \tau_1, \ldots, \tau_k)|}{|w_{bath}(k + 2, \tau_1, \ldots, \tau_k; \tau_i, \tau_j)|} \frac{((k + 2)/2)^2}{\beta^2}\right). \quad (3.14)$$

We remind the reader of the fact that the local weight w_{loc} in equation (3.13) and (3.14) is expressed relative to a factor $k + 4$, while the bath weight w_{bath} is expressed relative to a factor $k + 2$. The ambiguity comes from the two worm operators in the local trace without hybridization lines. Finally, we mention that an equal expression for worm inserts and removes with four times for the worm operators is found in $\mathcal{C}_{G^{(2)}}$.

3.2.3 Worm Replacement Step in $\mathcal{C}_{G^{(n)}}$

We have introduced operator shift moves in the previous chapter in order to decrease auto-correlation times. In the partition function space $\mathcal{C}_Z$ this move may be considered as an optional step. Sampling in $\mathcal{C}_Z$ will be ergodic, even if we ignore operator shifts, and just allow for pair inserts and pair removes. In worm sampling a shift step is not optional and is required for acceptable auto-correlation lengths. We will elaborate on this requirement here. Assume a local trace filled with hybridization operator pairs as the one illustrated in Figure 19. We now attempt to insert a worm pair into this trace. It turns out that inserting a worm pair, where the worm operators are relatively close to one another is probable, while inserting a worm pair where the worm operators are far apart is less probable. This is because quantum number violations may occur in the second case, while inserting a density pair (like in the first case) is usually possible.

The problem is especially severe for a large amount of operators in the trace, which occurs at small interaction or high temperatures. Additionally, very restrictive

53
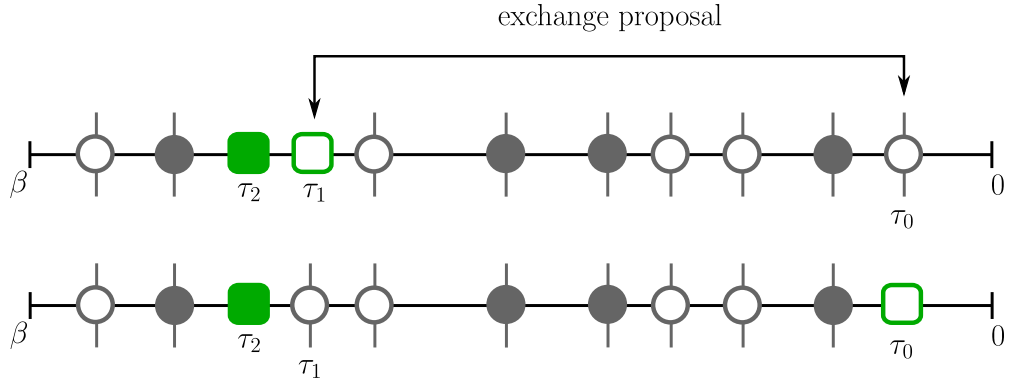
Figure 20: Illustration of the exchange procedure of a worm operator at τ_1 with an operator of the same flavor with hybridization line at τ_0. The local trace stays the same but the hybridization matrix has to be rebuild.

interaction types, such as the density-density interaction, produce more rejects due to quantum number violations of attempted inserts. This way we do not observe this auto-correlation problem at high temperatures, high interaction parameters and more general interactions such as Slater-Kanamori interactions (see Section 4.1.2 for results).

The solution to this problem is found in shift moves. In its more general form shift moves are computationally expensive, as we need to recalculate the local trace if we shift an operator from $\tau_i \to \tau_i'$. While calculating the time evolution is still very simple if the shift move does not jump over operators, it becomes more expensive if operators actually switch position. For this matter we will modify the concept of shift moves in the following. Instead of allowing for continuous shifts, we allow only for discrete shifts of worm operators in imaginary time. Specifically, the discrete shift refers to an exchange of a worm operator with an operator of the same flavor, which has a hybridization line connected as illustrated in Figure 20. This way we do not have to recalculate the local trace, as two locally indistinguishable operators switch position. Instead we need to recalculate the determinant of the hybridization matrix, as the exchange corresponds to a shift of the worm operator and a shift of the hybridization operator. Further we will not encounter any rejects of proposed moves due to local quantum number violations. It may seem that using exchange moves instead of proper shift moves is a limitation. We will show, however, that exchange move work specifically well for a very high number of operators with hybridization lines. This however is also the region, where we encounter the most severe auto-correlation length problems (i.e. ergodicity problems) due to quantum number violations.

It turns out, however, that worm replacement moves (or in the same way worm shift moves) are equally important for traces with very few operators. This occurs

54
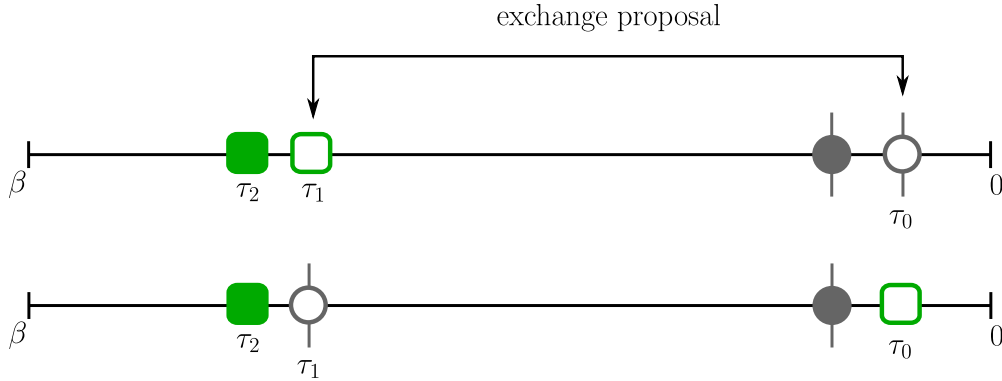
Figure 21: Illustration of the exchange procedure of a worm operator at τ_1 with an operator of the same flavor with hybridization line at τ_0 for a trace with few operators. The worm pair and the hybridization operator pair are inserted as density-like pairs. The replacement move generates a worm of length close to $\frac{\beta}{2}$.

if the interaction approaches the atomic limit. We are then restricted to inserting operator pairs into the trace, where the operators are very close to each other. These pairs have similar properties as density operators and can in principle be inserted for very high insulating cases. On the other hand, the high interaction strength does not allow for inserting operator pairs where τ is close to $\frac{\beta}{2}$.

Operator pairs, where τ is close to $\frac{\beta}{2}$ are not inserted due to both the local time evolution, which goes with $e^{-U\tau}$, and the strong insulating hybridization function. By inserting density-like pairs at short distances $\tau_i - \tau_j$ and then exchanging a worm operator with a hybridization operator we are able to pass this restrictions of the time evolution (Figure 21). The replacement move only depends on the ratio of the determinant of the hybridization matrix. We will show this in the following.

The proposal probability is similar to the proposal probability of a shift move:

$$f'(\mathcal{C}_{G^{(1)}}, k \to k) = \frac{1}{2(k/2)}. \tag{3.15}$$

This corresponds to selecting one of the two worm operators at random and selecting one operator of the same flavor with a hybridization line (we choose an operator from the $k/2$ operators of the same type (annihilator/creator) and then discard flavors, which are not equivalent to the worm flavor). The proposal probability of switching the operators back to their original position is given by:

$$f(\mathcal{C}_{G^{(1)}}, k \to k) = \frac{1}{2(k/2)}. \tag{3.16}$$

We observe that the proposal probabilities for the replacement move cancel out

and the acceptance ratio is fully determined by the ratio of weights:

$$\frac{a'(\mathcal{C}_{G^{(1)}}, k \rightarrow k)}{a(\mathcal{C}_{G^{(1)}}, k \rightarrow k)} = \frac{|w_{loc}(k+2, \tau_1, \ldots, \tau_i, \ldots, \tau_j, \ldots, \tau_k) w_{bath}(k, \tau_1, \ldots, \tau_i, \ldots, \tau_k)|}{|w_{loc}(k+2, \tau_1, \ldots, \tau_i, \ldots, \tau_j, \ldots, \tau_k) w_{bath}(k, \tau_1, \ldots, \tau_j, \ldots, \tau_k)|},$$
(3.17)

where τ_i refers to the initial position of the worm and τ_j to the initial position of the operator with the hybridization line. The replacement move is fully determined by the determinant of the hybridization matrix as the local weight cancels. The Metropolis acceptance rate is given by:

$$a'(\mathcal{C}_{G^{(1)}}, k \rightarrow k) = min\left(1, \frac{|w_{bath}(k, \tau_1, \ldots, \tau_i, \ldots, \tau_k)|}{|w_{bath}(k, \tau_1, \ldots, \tau_j, \ldots, \tau_k)|}\right).$$
(3.18)

Up to this point we have stated that shift moves and replacement moves are somewhat similar. We would like to use the opportunity to point out the difference between a worm replacement and a worm shift move. The acceptance rate of the worm replacement move depends on a determinant ratio of two matrices of dimension $(k/2 \times k/2)$, where k here refers to the number of operators with hybridization lines connected. In that sense it is very comparable to the determinant ratio in estimator of partition function sampling of two matrices of dimension $(k/2 - 1 \times k/2 - 1)$ and $(k/2 \times k/2)$ in equation (3.1). The acceptance rate of a worm shift move, on the other hand, only depends on the ratio of the local traces. While for the worm replacement move we are able to pass the restrictions of the local time evolution, for the worm shift move we are able to pass the restrictions of the hybridization function. When calculating strong insulating cases we profit the most if we consider both moves. Implementing a proper shift move is left however for the future.

3.3   Worm Measurement

We will now show how the measurement of Green's function looks in $\mathcal{C}_{G^{(n)}}$. It turns out that the measurement itself is trivial and we only need to find the correct normalization of the Green's functions measured and the correct sign. For the one-particle Green's function $G^{(1)}$ a worm is defined by the operators $d(\tau_i)$ and $d^\dagger(\tau_j)$. The correct weight is intrinsically given as we sample in the Green's function space $\mathcal{C}_{G^{(n)}}$. Thus, the estimator of the Green's function simply follows as:

$$G^{(1)}_{\mathcal{C}_G}(\tau) = \langle \text{sgn} \cdot \delta(\tau, \tau_i - \tau_j) \rangle_{MC}.$$
(3.19)

The Green's function in Matsubara frequencies can be calculated by substituting the δ-function by the Fourier transform:

$$G^{(1)}_{\mathcal{C}_G}(i\omega) = \langle \text{sgn} \cdot e^{i\omega(\tau_i - \tau_j)} \rangle_{MC}.$$
(3.20)
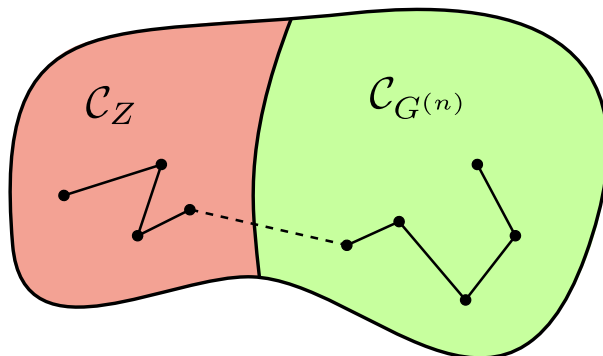
Figure 22: Illustration of a Markov chain in the configuration spaces $\mathcal{C}_Z$ and $\mathcal{C}_{G^{(n)}}$. We extract the normalization factor $\frac{1}{Z}$ for the Green's functions by counting how many steps are made in $\mathcal{C}_Z$.

Both expressions 3.19 and 3.20 are not normalized to Z, but rather $Z_{G^{(1)}}$ as defined in equation (3.2). The measurement of the two-particle Green's function $G^{(2)}$ is done in the same way. We will now discuss the normalization and the sign in more detail.

3.3.1 Normalization

For now we have not given any reason why we need to sample in $\mathcal{C}_Z$. One could assume, that measuring in $\mathcal{C}_{G^{(n)}}$ is enough and we do not need to move back and forth between $\mathcal{C}_{G^{(n)}}$ and $\mathcal{C}_Z$. This is a fair assumption, since in principle we are ergodic in $\mathcal{G}^{(n)}$, when assuming Worm replacement or Worm shift moves. It turns out that we need to sample both in $\mathcal{C}_{G^{(n)}}$ and $\mathcal{C}_Z$ with about the same number of steps to fix our normalization, as the thermal expectation value is given by:

$$\langle \mathcal{O}(\tau) \rangle = \frac{1}{Z} Tr \left(T_\tau e^{-\beta H} \mathcal{O}(\tau) \right). \tag{3.21}$$

As we are not sampling in the partition functions space $\mathcal{C}_Z$ any more, we need to consider the factor $\frac{1}{Z}$ explicitly. We do this by sampling in both spaces. We illustrate this in Figure 22.

When measuring the Green's functions in $\mathcal{C}_{G^{(n)}}$ we implicitly normalize with the number of steps taken in $\mathcal{C}_{G^{(n)}}$. We correct for this factor by explicitly counting how many steps N_G were taken in $\mathcal{C}_{G^{(n)}}$. We further count how many steps N_Z were taken in $\mathcal{C}_Z$. This estimates the size of the configuration space $\mathcal{C}_Z$, which then gives the correct normalization. The normalization for $G^{(n)}$ is then given by [41]:

$$G^{(n)} = \frac{1}{\eta} \frac{N_G}{N_Z} G^{(n)}_{\mathcal{C}_G}, \tag{3.22}$$

57

where $G_{\mathcal{C}_G}^{(n)}$ is measured in $\mathcal{C}_{G^{(n)}}$ and the factor $1/\eta$ is a result of rescaling $Z_{G^{(n)}}$ in equation (3.3).

We point out that in order to calculate the Monte Carlo expectation value, we still need to divide the result by the number of measurements N taken. It is important to notice the difference between the number of measurements N and the number of steps N_G and N_Z taken. We observe the difference when dealing with auto-correlation lengths greater than a single Monte Carlo step (which is usually always the case). The auto-correlation length in worm space $\mathcal{C}_{G^{(n)}}$ looks very different from the auto-correlation in partition function space $\mathcal{C}_Z$. We need a more elaborate scheme to consider the auto-correlation length in worm space. Simply skipping an amount of steps close to the auto-correlation length before measuring the configuration again, as it is done in partition function sampling, may ignore certain worm configurations fully. We leave this task for the future.

Before turning to the discussion of the worm sign, we mention that equation (3.22) is only one way of normalizing the worm measurement. In a different scenario, we could imagine to do the entire sampling in worm space, without removing the worm operators at all. We are then required to generate worm configurations by shift moves and replacement moves. In this case, we would need to normalize the result by assuming some physical knowledge of the Green's functions behavior. One possibility is to extract the normalization by assuming the correct behavior of the asymptotics of $G^{(1)}(i\omega)$ and $G^{(2)}(i\nu, i\nu, i\omega)$.

3.3.2   Worm Sign

We have already discussed the QMC sign in the previous chapter for the hybridization expansion. We will now return to this discussion once more, as we have now introduced worm operators into our local trace. These operators are local operators, which contribute to the sign of the configuration and need to be considered carefully. We will now give a short summary of how the existing sign is changed by the worm operators.

Sign of the local time ordered Trace s_{loc}
From a local viewpoint we are not able to distinguish worm operators and operators with a hybridization line connected. As a result the time evolution of the local time ordered trace may in principle generate a sign. We make the reader aware of this sign as the contribution from the worm may not be compensated by the hybridization function.

Sign of ordering of Trace into alternating Pairs s_{alt}
We remind the reader of the sign contribution from ordering the local trace into alternating pairs. This accounts for the fact that the hybridization matrix only

connects annihilation to creation operators and not annihilation to annihilation operators or creation to creation operators. The worm operators are not included in this sorting procedure and need to be sorted to the beginning or to the end of the trace. This way the sorting into alternating pairs of creation and annihilation operators is not changed by the occurrence of worm operators in the trace.

4   Results

In this chapter we will present the results of calculating $G^{(1)}$ (in imaginary time and Matsubara frequencies) and $G^{(2)}$ (in Matsubara frequencies) with the worm algorithm. We will compare the outcome to the results found using partition function sampling. We observe that the two methods agree very well with one another.

We calculate the results on the Anderson impurity model and DMFT Bethe lattice with an energy scale determined by the half-bandwidth $D = 1$ (unless otherwise stated). Different interaction strengths (from metallic to insulating) and interaction types (density-density and Slater-Kanamori) and different temperatures are considered. Calculations are generally done out of half-filling Moving away from half-filling further reveals differences between worm sampling and sampling in partition function space.

4.1   One-Particle Green's Function $G^{(1)}$

In this section we summarize all results of $G^{(1)}$. As reference, we always supply the results of $G^{(1)}$ measured in partition function space as well. The results are presented in the following way: first we establish by what means we are allowed to compare the measurement of $G^{(1)}$ made in worm space and in partition function space. Then we revisit the ergodicity problem in worm space solved by worm replacement steps. We do this using single-shot QMC calculations. Following this, we benchmark the worm algorithm with reference to the measurement in partition function space in the metallic case and close to the Mott metal-insulator transition. This will require us to use the DMFT self-consistency loop. Once we have shown that the worm measurement gives correct results, we show how it performs for strong insulating systems approaching the atomic limit. Lastly, we show results of how the worm algorithm performs for more orbitals and more general interaction types.

4.1.1 Simulation Time and Measurement Steps

In order to explore the strengths and weaknesses of the worm algorithm over conventional methods of CT-Hyb, we are required to look at the differences in the errorbars generated. When measuring $G^{(1)}(\tau)$ in the partition function space the estimator takes on the form:

$$G_{\mathcal{C}_Z}^{(1)}(\tau) = \langle \sum_{ij} \frac{\det \boldsymbol{\Delta}^{(ij)}}{\det \boldsymbol{\Delta}} \text{sgn} \cdot \delta(\tau, \tau_i - \tau_j) \rangle_{MC}, \qquad (4.1)$$

where $\Delta^{(ij)}$ is the hybridization matrix with the i-th row and j-th column removed. This means, that we measure $(k/2)^2$ contributions to the Green's function during each measurement step, where $k/2$ is the expansion order.

When measuring $G^{(1)}(\tau)$ in Green's function space, the estimator takes on the form:

$$G_{\mathcal{C}_G}^{(1)}(\tau) = \langle \text{sgn} \cdot \delta(\tau, \tau_i - \tau_j) \rangle_{MC}. \qquad (4.2)$$

This means, that we measure only one contribution to the Green's function during each measurement step. We observe that the measurement in partition function space includes more information for higher expansion orders $k/2$ than the measurement in worm space. However, the measurement in partition function space also takes significantly longer for higher expansion orders as we have to loop over all possible combinations of operators (computing the determinant ratio is not an issue due to the fast determinant updates by inverse by partitioning).

In Figure 23 we show the differences in the errorbars of worm sampling and partition function sampling for the Anderson impurity model with a single-orbital Bethe lattice as bath for an equal number of measurement steps on a logarithmic scale. The interaction is set to density-density with $U = 0.5D$, the chemical potential to $\mu = 0.3D$ and the inverse temperature to $\beta = 200/D$. The parameters are chosen such that the expansion order is high ($k/2 \sim 40$). The parameter η is set to $\eta = 0.15/\beta$, which assures that we spend an equal amount of time in $\mathcal{C}_Z$ and $\mathcal{C}_{G^{(1)}}$.

For an equal amount of measurement steps in worm sampling and partition function sampling we observe smaller error bars in the measurement made in partition function space. This can be explained by the high expansion order. The estimator in partition function space given by equation (4.1) considers all $(k/2)^2$ possible pairs at each measurement step. The estimator in worm space given by equation (4.2) only considers one operator pair (i.e. the worm) at each measurement step. In Figure 24 we repeated the measurement with the same parameters. Instead of fixing the number of measurement steps, we fix the simulation time. The worm measurement is faster than the partition function measurement. We observe that the errorbars are comparable for a similar simulation time. We conclude that
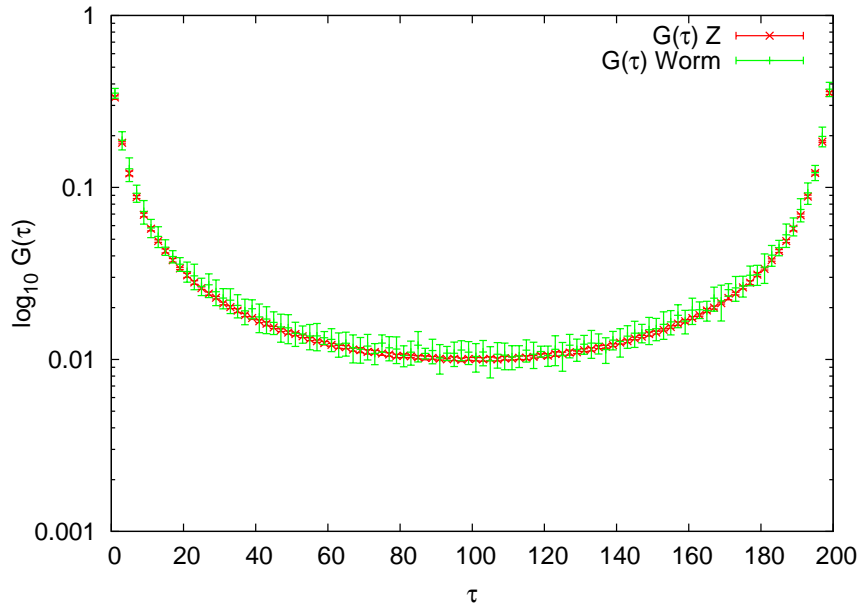
Figure 23: Errorbars of $G^{(1)}_{\uparrow\uparrow}(\tau)$ on a logarithmic y-scale using partition function sampling (red crosses) and worm sampling (green pluses) for an equal amount of measurement steps in each space.
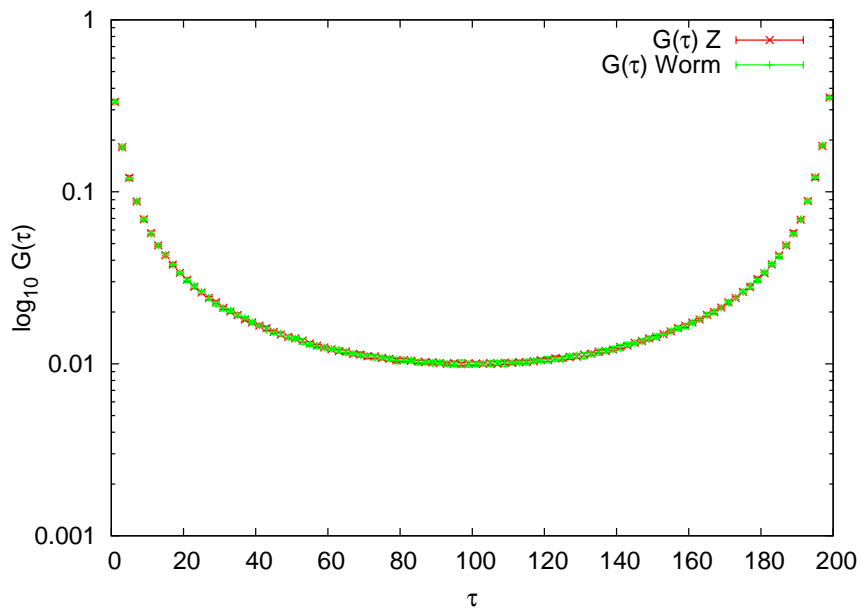
Figure 24: Errorbars of $G^{(1)}_{\uparrow\uparrow}(\tau)$ on a logarithmic y-scale using partition function sampling (red crosses) and worm sampling (green pluses) for a similar simulation time in each space.

we are able to compare the results of the worm algorithm to the results of the conventional method in CT-Hyb by assuming equal simulation times. In the above calculation we enabled worm replacement steps. In the next section we will show why worm replacement steps are necessary for ergodicity.

4.1.2 Worm Replacement Steps

In the previous chapter we have discussed some ergodicity problems of worm sampling. For high expansion orders it is difficult to insert worms, where operators are separated by a length close to $\beta/2$. In Figure 25 we show this problem. We have used the same parameters to generate high expansion orders as in the previous section 4.1.1. We use a density-density interaction with $U = 0.5D$, $\mu = 0.3D$ and $\beta = 200/D$. The average expansion order is $k/2 \sim 40$. We have disabled replacement steps and adjusted η to $\eta = 0.22/\beta$. This assures that we spend an equal amount of steps in each space.

We observe that we are restricted to inserting worm operators, which are very close to one another. At around $\tau = 30$ and $\tau = 170$ the auto-correlation (ergodicity) problem kicks in and the error in the estimator becomes very large (see Figure 25). The asymmetry between the fluctuations around $\tau = 30$ and $\tau = 170$ follows from the fact that the calculation is done out of half-filling. In between this interval we are not able to insert any worms into the trace. In Figure 26 we propose worm replacement moves with a probability of 0.3 at $\eta = 0.15/\beta$. We observe that the auto-correlation (ergodicity) problem vanishes and that we are now able to generate worm operators with a length close to $\beta/2$.
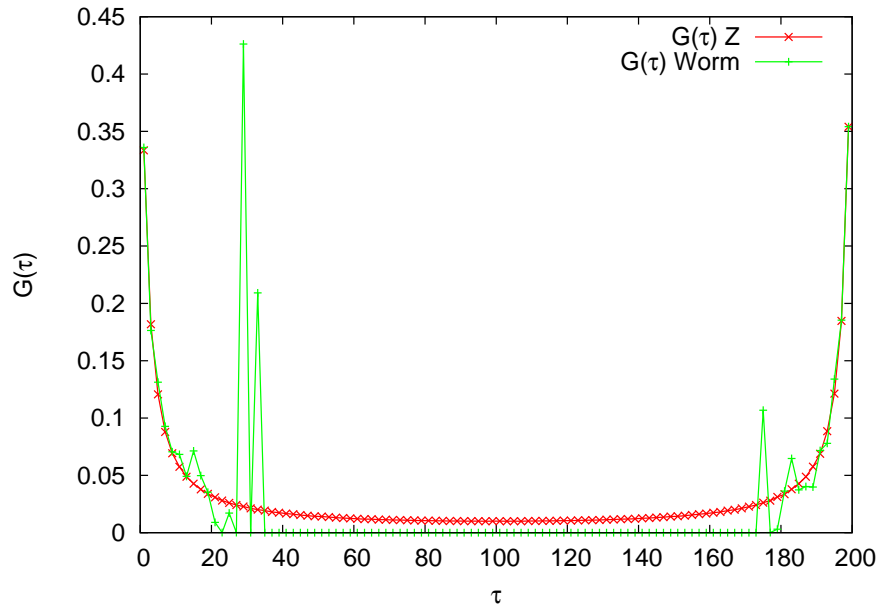
Figure 25: $G_{\uparrow\uparrow}^{(1)}(\tau)$ (without errorbars) using partition function sampling (red crosses) and worm sampling (green pluses) without replacement steps.
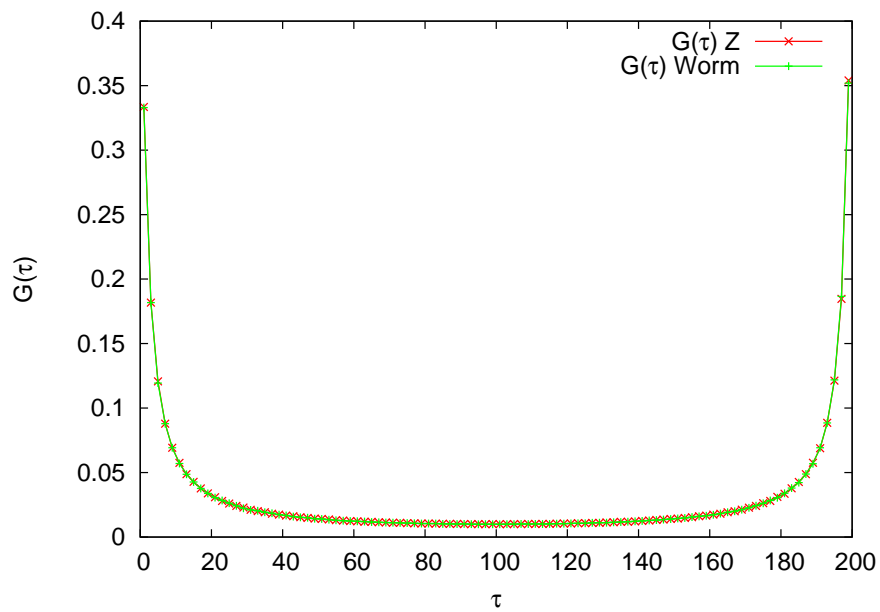
Figure 26: $G_{\uparrow\uparrow}^{(1)}(\tau)$ (without errorbars) using partition function sampling (red crosses) and worm sampling (green pluses) with replacement steps.

4.1.3 Metallic Systems

We now show how the worm algorithm performs in the DMFT loop. First we calculate a metallic system with $U = 1.0D$, $\mu = 0.6D$, $\beta = 50/D$ for the single-orbital Bethe lattice. We use 10 DMFT self-consistency steps and check for convergence by looking at the self-energy. For the worm algorithm we further set the probability of worm inserts and removes to 0.1 and the probability of worm replacement moves to 0.2. The remaining probability is equally distributed for inserts and removes of operator pairs with hybridization lines. By setting $\eta = 2.2 \cdot 10^{-1}/\beta$, we assure that we spend an equal amount of steps in each space.

In Figure 27 we show how $G^{(1)}$ looks after the DMFT self-consistency. The errorbars for worm sampling and partition function sampling look comparable. Smaller errorbars in partition function sampling are a consequence of using the Fourier transform of a Legendre filtered $G^{(1)}$ for the self consistency loop [29].

When looking at the momentum integrated spectral function $A(\omega)$ in Figure 28 we find good agreement between the result from sampling $G^{(1)}$ in worm space and in partition function space. The single well-defined peak characterizes the system as being metallic. Small differences in $A(\omega)$ are a consequence of the differences in the errorbars in $G^{(1)}$.
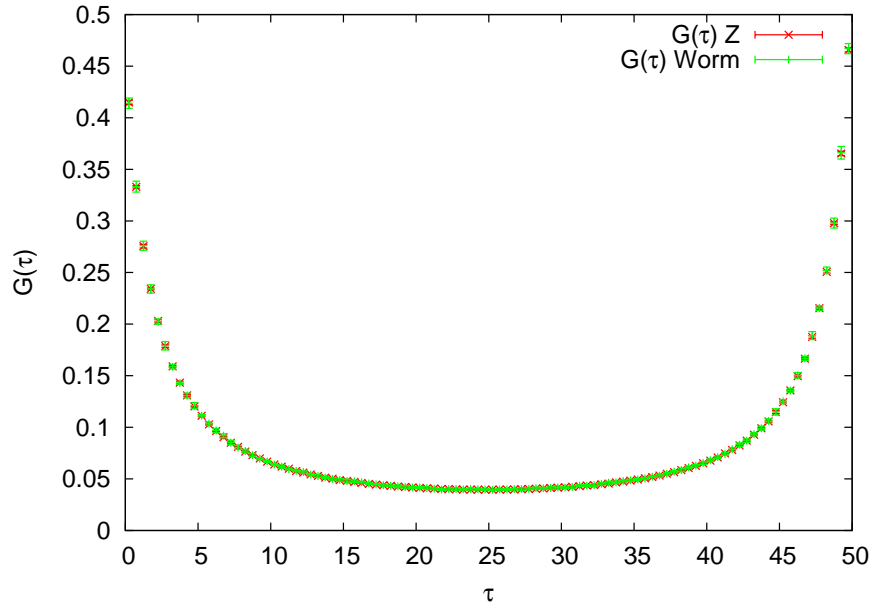
Figure 27: Metallic $G_{\uparrow\uparrow}^{(1)}(\tau)$ with errorbars using partition function sampling (red crosses) and worm sampling (green pluses) after 10 DMFT steps.
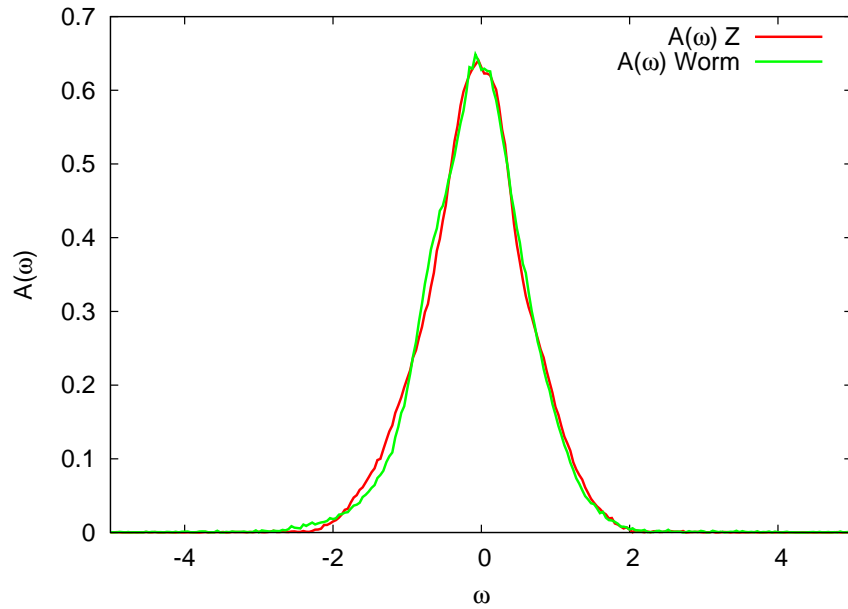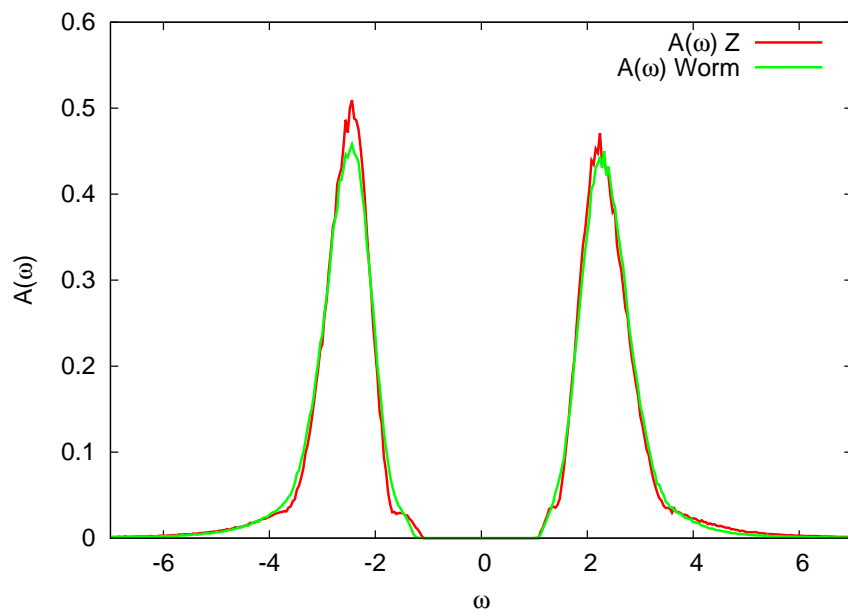

Figure 28: $A(\omega)$ resulting from analytic continuation of $G^{(1)}(\tau)$ in Figure 27.

4.1.4 Systems close to the Mott Metal-Insulator Transition

We will now continue to benchmark the worm algorithm in the Mott metal-insulator transition. The parameters of the system we look at are given by $U = 2.4D$, $\mu = 1.3D$ and $\beta = 50/D$. Again, we use 10 DMFT self-consistency steps and check for convergence by looking at the self-energy. For the worm algorithm we set the probability of worm inserts and removes to 0.1 and the probability of worm replacement moves to 0.2. The remaining probability is equally distributed for inserts and removes of operator pairs with hybridization lines. By setting $\eta = 3.0 \cdot 10^{-1}/\beta$, we assure that we spend an equal amount of steps in each space.

When looking at $G_{\uparrow\uparrow}^{(1)}(\tau)$ in Figure 29 we find good agreement between the results from sampling in worm space and sampling in partition function space. We also find that the errorbars are in good agreement for the two Green's functions. From this data we produce the momentum integrated spectral function $A(\omega)$ in Figure 30. The two distinct peaks forming correspond to the Hubbard bands, in between there is a sharp Kondo peak. When looking at the spectral function $A(\omega)$ we can see that the system is still on the metallic side, but the sharpness of the Kondo peak indicates that the Mott metal-insulator transition occurs for a slightly larger U. Sampling in worm space and partition function space gives a comparable result.
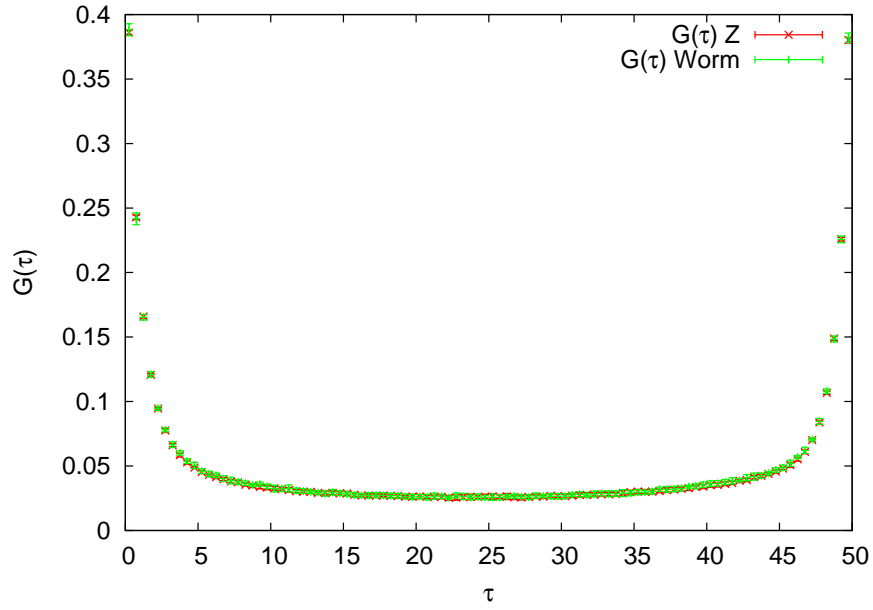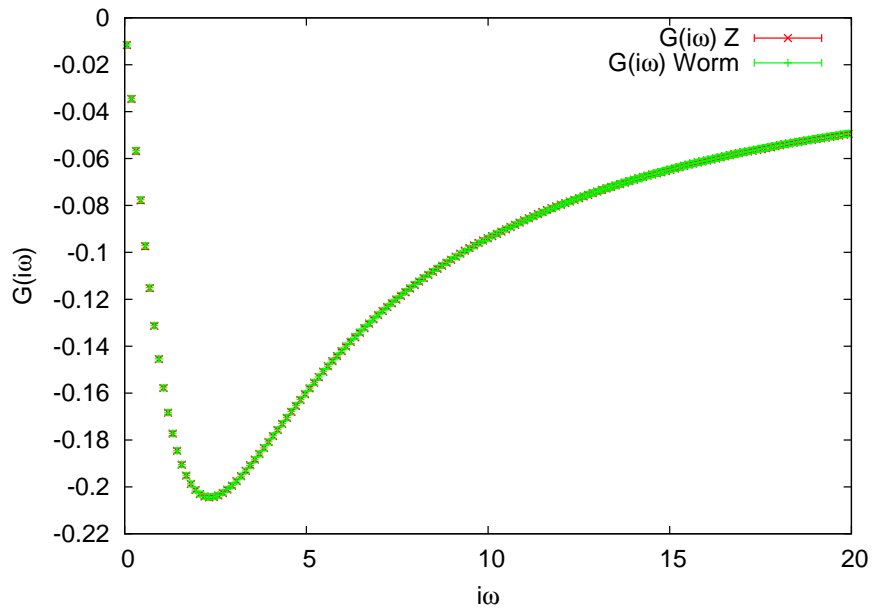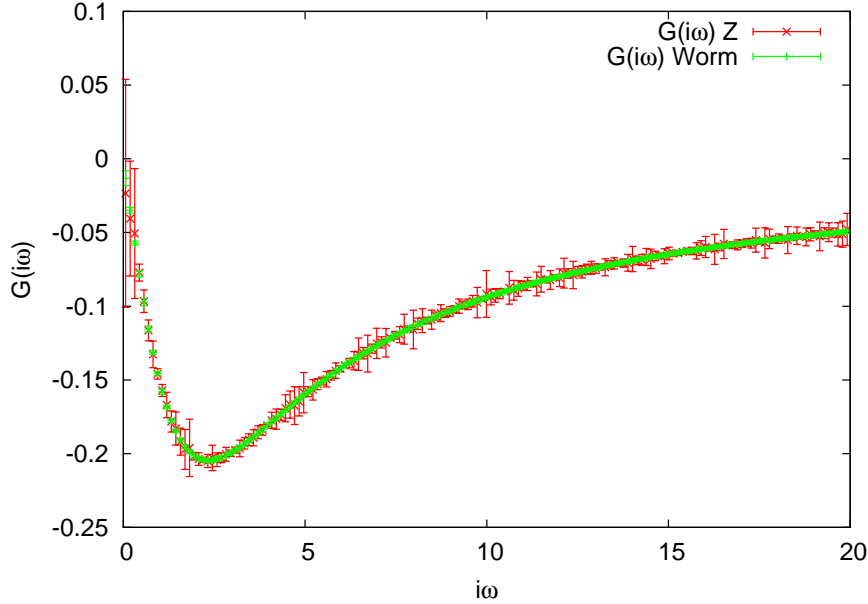
Figure 29: $G_{\uparrow\uparrow}^{(1)}(\tau)$ for the MIT with errorbars using partition function sampling (red crosses) and worm sampling (green pluses) after 10 DMFT steps.
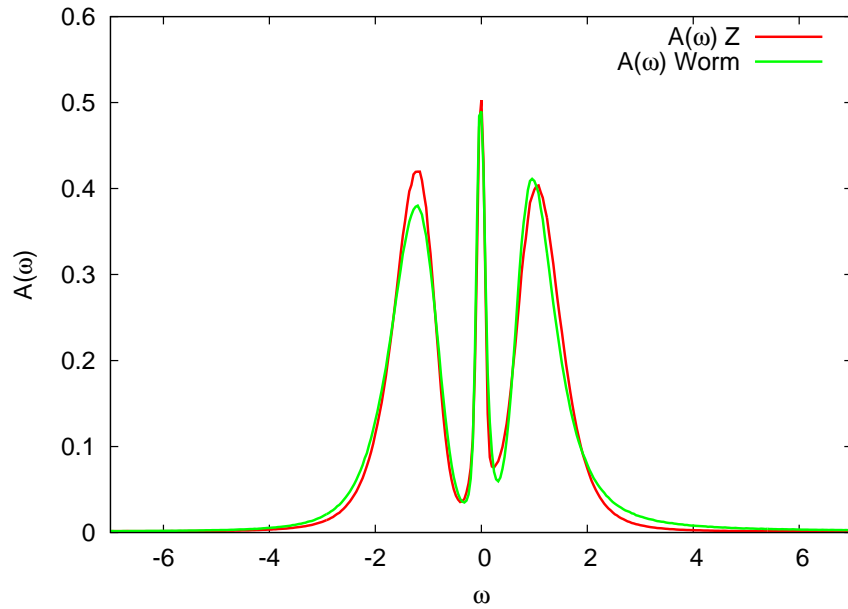
Figure 30: $A(\omega)$ resulting from analytic continuation of $G^{(1)}(\tau)$ in Figure 29.

4.1.5 Approaching the Atomic Limit

We now show how the worm algorithm performs in the strong insulating case. We set the parameters of the system to $U = 5.0D$, $\mu = 2.6D$ and $\beta = 50/D$ for the single-orbital Bethe lattice. We use 10 DMFT self-consistency steps, expecting to find a Mott insulator. At this point we will no longer use the Legendre filtered $G^{(1)}(\tau)$ for the Green's function measurement in partition function space, which then enters the DMFT loop after Fourier transformation, but rather use the direct Fourier transform of $G^{(1)}(\tau)$. While for the metallic case we do not expect any differences between the Legendre filtered Green's function and the measurement through binning, we find artificial periodic structures for the strong-insulating case on a logarithmic scale, which we want to avoid. For the worm algorithm we set the probability of worm inserts and removes to 0.2 and the probability of worm replacement moves to 0.3 and $\eta = 1.4/\beta$. We choose a slightly higher probability for worm replacement moves with respect to the previous runs, since the histogram of the average expansion order is more localized. Due to higher acceptance rates, it is better to allow for more replacement moves than insert moves.

In Figure 31 we show the strong insulating Green's function $G^{(1)}(\tau)$ for the above parameters. We see good agreement between the measurement in partition function space and worm space. When viewing the momentum integrated spectral function $A(\omega)$ in Figure 32, we find two distinct peaks separated by U, resulting in a Mott insulator.

In Figure 33 we show the strong insulating Green's function $G^{(1)}(\tau)$ on a logarithmic y-scale. We observe that both, worm sampling and sampling in partition function space, produce fluctuating errorbars. This can be explained by the fact that the hybridization function does not allow for any operator pairs to be inserted near $\beta/2$. Also in the region, where the second derivative of the hybridization function changes the most, larger errorbars can be observed. On first sight, the errorbars of worm sampling look worse than the errorbars of partition function sampling. However, we assume that the errorbars in partition function space towards $\beta/2$ are too small and are a result of ergodicity problems of the estimator.

In Figure 34, the Green's function $\Im(G^{(1)}_{\uparrow\uparrow}(i\omega))$ is shown as a function of Matsubara frequencies without possible discretization errors. We find good agreement between worm sampling and sampling in partition function space. We observe that the worm algorithm converges faster than the conventional algorithm. In Figure 35, we show $\Im(G^{(1)}_{\uparrow\uparrow}(i\omega))$ after 4 DMFT iterations. At this point the self-energy jumps quite rapidly.
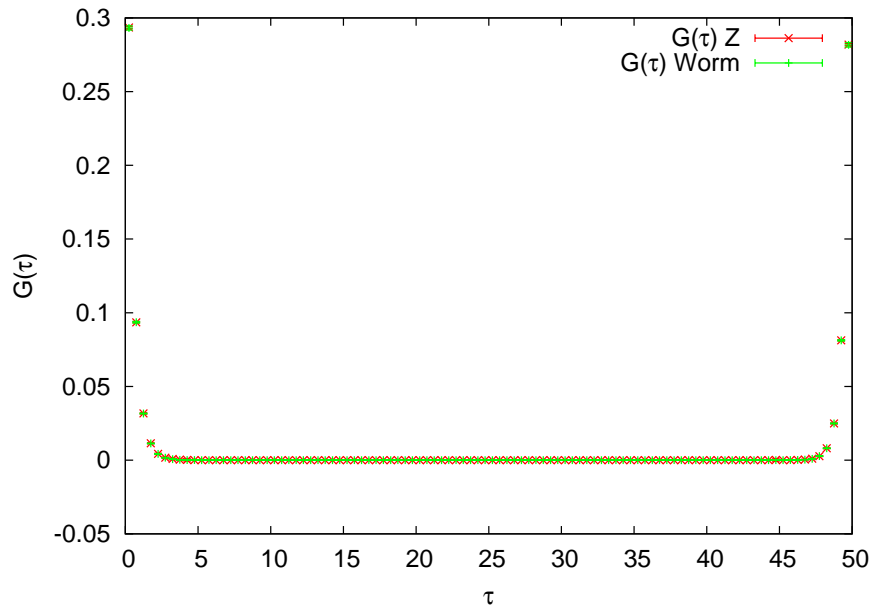
Figure 31: $G^{(1)}_{\uparrow\uparrow}(\tau)$ for the strong insulating case with errorbars using partition function sampling (red crosses) and worm sampling (green pluses) after 10 DMFT steps.

Figure 32: $A(\omega)$ resulting from analytic continuation of $G^{(1)}(\tau)$ in Figure 31.
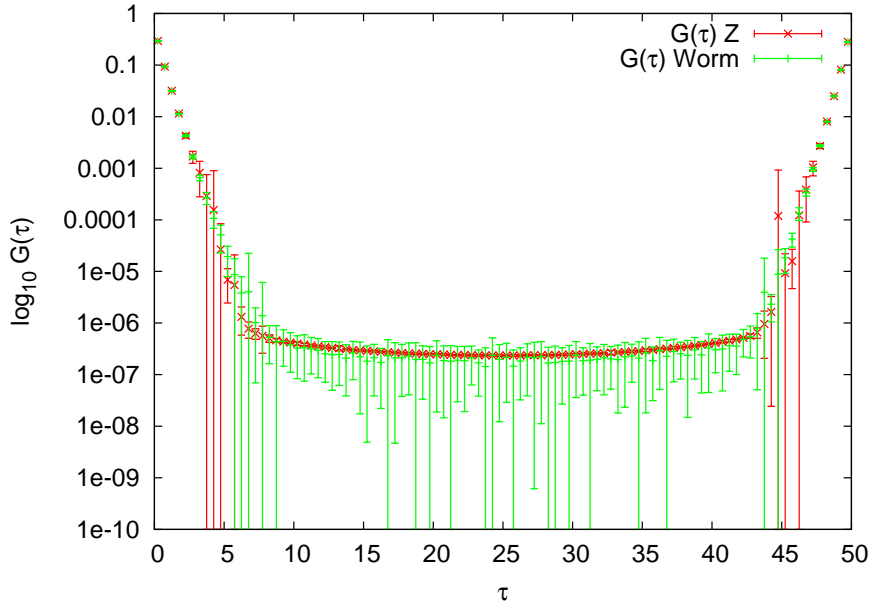
Figure 33: $G_{\uparrow\uparrow}^{(1)}(\tau)$ errorbars of Figure 31 on a logarithmic y-scale for the strong insulating case using partition function sampling (red crosses) and worm sampling (green pluses) after 10 DMFT steps.

Figure 34: $\Im(G_{\uparrow\uparrow}^{(1)}(i\omega))$ for the strong insulating case with errorbars using partition function sampling (red crosses) and worm sampling (green pluses) after 10 DMFT steps.

Figure 35: $\Im(G^{(1)}_{\uparrow\uparrow}(i\omega))$ for the strong insulating case with errorbars using partition function sampling (red crosses) and worm sampling (green pluses) after 4 DMFT steps.

4.1.6 Multi-Orbital Slater-Kanamori Results

In this section we present some results from sampling the Green's function $G^{(1)}$ of a multi-orbital system with Slater-Kanamori interaction parameters. The Slater-Kanamori Hamiltonian was already presented in Equation (1.8). We will calculate a two-orbital Bethe lattice with two different bandwidths. We set the parameters according to Reference [27], where the same calculation has already been made. That is, we set $D_1 = 1.0$ for the first orbital and $D_2 = 2.0$ for the second orbital. We choose $U = 3.0D_1$ and $J = 0.75D_1$, and use $U' = U - 2J = 1.5D_1$. The inverse temperature is set to $\beta = 100/D_1$. The half-filling condition for the Slater-Kanamori case is given by $\mu = \frac{3}{2}U - \frac{5}{2}J$ resulting in $\mu = 2.625D_1$. For the worm algorithm we set the probability of worm inserts and removes to 0.1 and the probability of worm replacement moves to 0.2. By setting $\eta = 3.25 \cdot 10^{-1}/(2\beta)$, we assure that we spend an equal amount of steps in each space (the factor 2 is an attempt to scale η with the number of orbitals). During the discussion of the results of the two-particle Green's function obtained using worm sampling in the next section, we suggest defining two different η-values for this system, since the two bands are quite different.

71
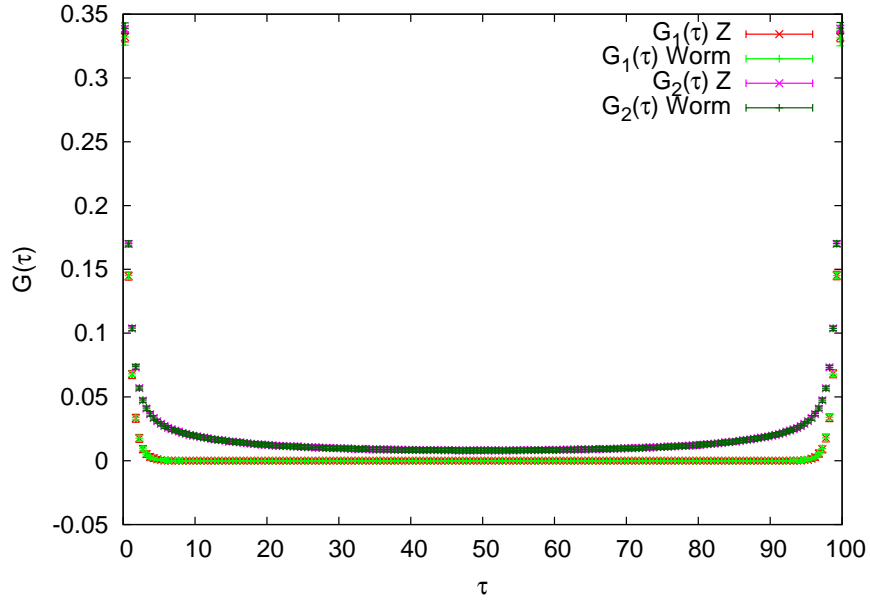
Figure 36: $G^{(1)}_{\uparrow\uparrow,1}(\tau)$ (first orbital), $G^{(1)}_{\uparrow\uparrow,2}(\tau)$ (second orbital) with errorbars for the Slater-Kanamori interaction using partition function sampling and worm sampling after 10 DMFT steps.
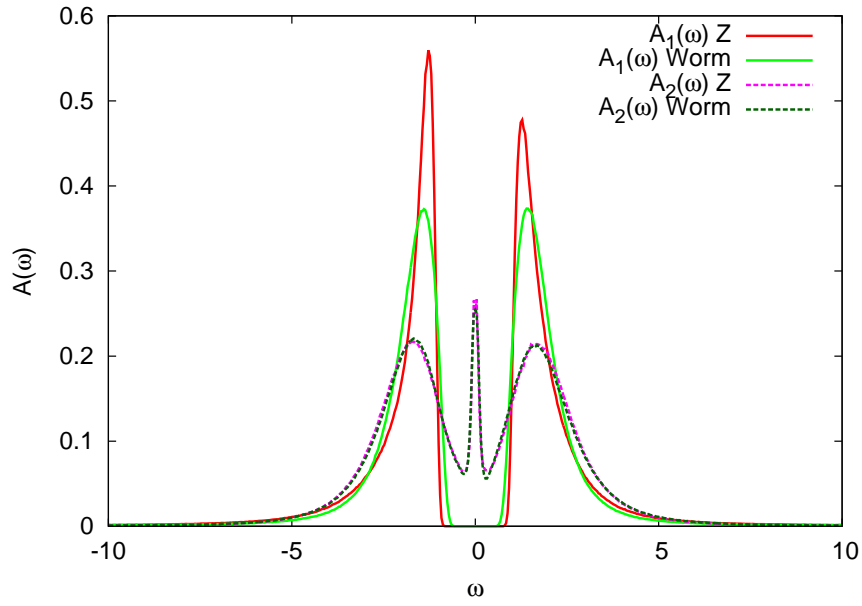
Figure 37: $A(\omega)$ resulting from analytic continuation of $G^{(1)}_{\uparrow\uparrow,1}$ (first orbital) and $G^{(1)}_{\uparrow\uparrow,2}(\tau)$ (second orbital) in Figure 36 using partition function sampling and worm sampling.

In Figure 36 we observe a strong insulating Green's function $G^1(\tau)$ for the second orbital. We also see that the first orbital is less insulating. However from Figure 36, we cannot verify if the first orbital is metallic or undergoing a transition. In Figure 37 we supply the momentum integrated spectral function for the two orbitals. We can now verify that the first orbital is metallic, close to the Mott metal-insulator transition, while the second orbital is clearly isolating. Worm sampling and sampling in partition function space give very similar results for both orbitals.

We observe differences in the height of the peaks of the insulating orbital of the momentum integrated spectral function $A(\omega)$ between worm sampling and sampling in partition function space. These deviations result from variations of the errorbars of $G^1(\tau)$. A better spectral function can be extracted with the Maximum Entropy method, when repeating this run with higher statistics. The momentum integrated spectral function of the worm result is more credible, since the two peeks are symmetric, which is what we expect in the half-filled case.

4.2 Two-Particle Green's Function $G^{(2)}$

We now show the results of measuring the two particle Green's function $G^{(2)}$ using worm sampling and sampling in partition function space. The results are presented in the following way: we will measure the two-particle Green's function $G^{(2)}$ for a metallic and a strongly insulating single-orbital Bethe lattice. Afterwards, we will measure $G^{(2)}$ for a two-orbital Bethe lattice with Slater-Kanamori interaction. All measurements in this section are based on converged DMFT results.

4.2.1 Metallic Systems

First, we calculate the two-particle Green's function $G^{(2)}$ for the converged DMFT results of the metallic case introduced in the previous section. The system is defined by $U = 1.0D$, $\mu = 0.6D$, $\beta = 50/D$. For the worm algorithm we set the probability of worm inserts and removes to 0.1 and the probability of worm replacement moves to 0.2 and $\eta = 7.5 \cdot 10^{-4}/\beta$. The Green's function is calculated for 80 fermionic Matsubara frequencies $i\nu, i\nu' = [-79\frac{\pi}{\beta}, 79\frac{\pi}{\beta}]$ and one bosonic Matsubara frequency $i\omega = 0$. In order to compare the errorbars of $G^{(2)}$ in worm sampling to the measurements in partition function space we need to produce two-dimensional slices of $G^{(2)}$, which we do by fixing two of three Matsubara frequencies. In Figure 38 we have fixed $i\omega = 0$. We illustrate how fixing the fermionic Matsubara frequency $i\nu'$ gives us a single slice of $G^{(2)}$.

When looking at the slices $\Re(G^{(2)}_{\uparrow\uparrow\uparrow\uparrow}(i\nu, \frac{\pi}{\beta}, 0))$ in Figure 39 and $\Re(G^{(2)}_{\uparrow\uparrow\downarrow\downarrow}(i\nu, \frac{\pi}{\beta}, 0))$ in Figure 40 we observe that worm sampling and sampling in partition function space are in good agreement for the metallic case.
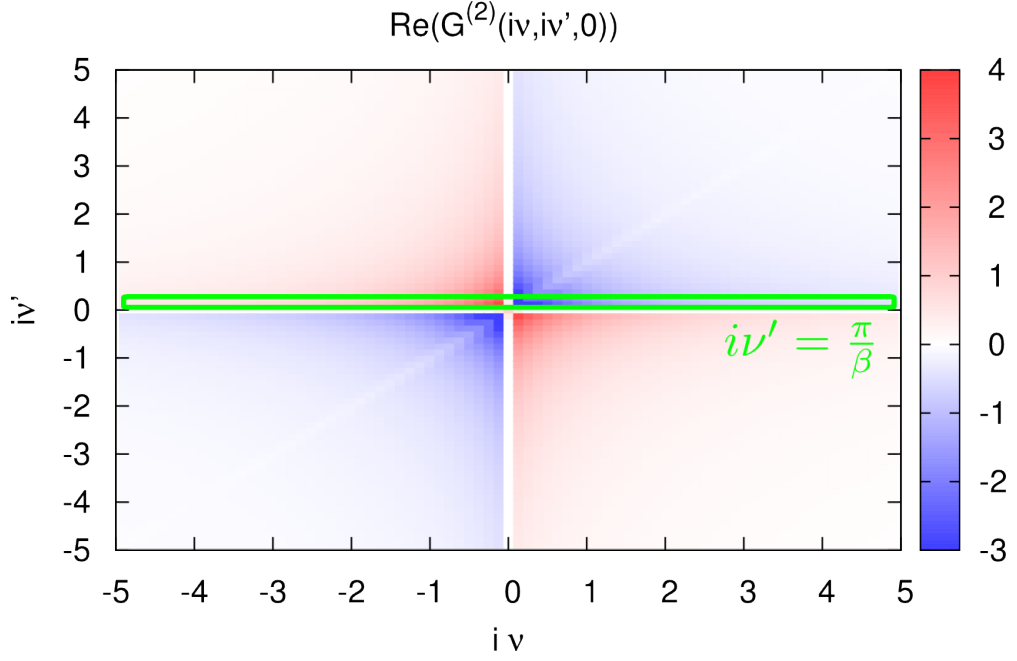
Figure 38: $\Re(G^{(2)}_{\uparrow\uparrow\uparrow\uparrow}(i\nu, i\nu', i\omega = 0))$ in the particle-hole convention with the amplitude encoded in false colors according to the bar at the right-hand side. A slice at the fermionic frequency $i\nu' = \frac{\pi}{\beta}$ is marked in green.
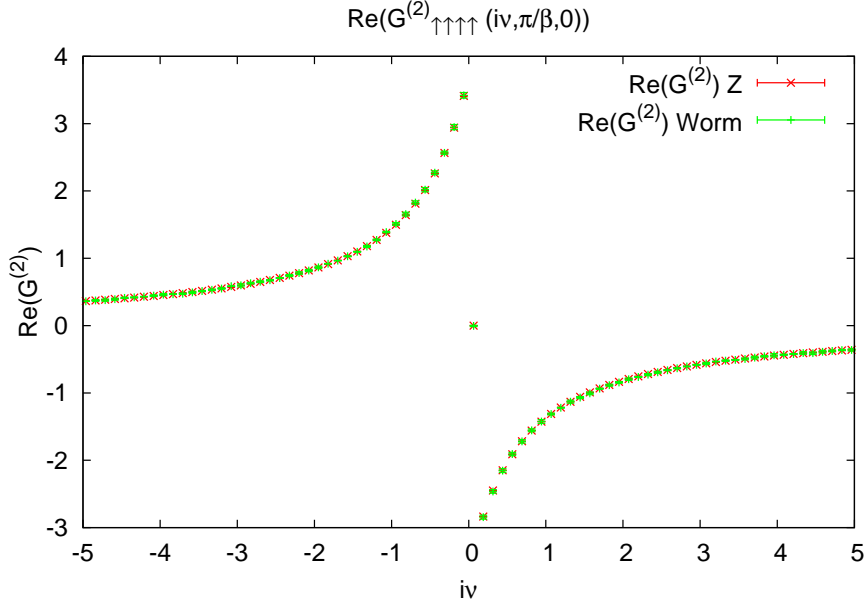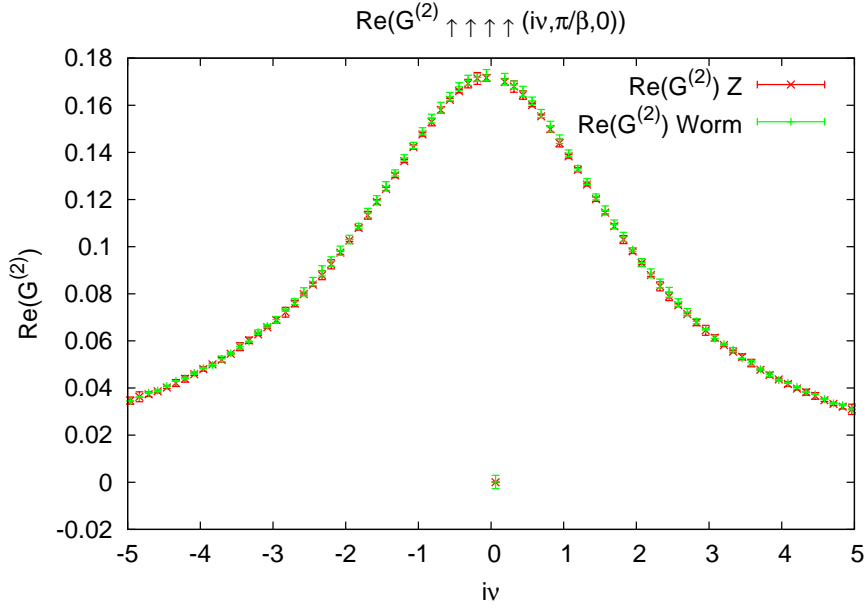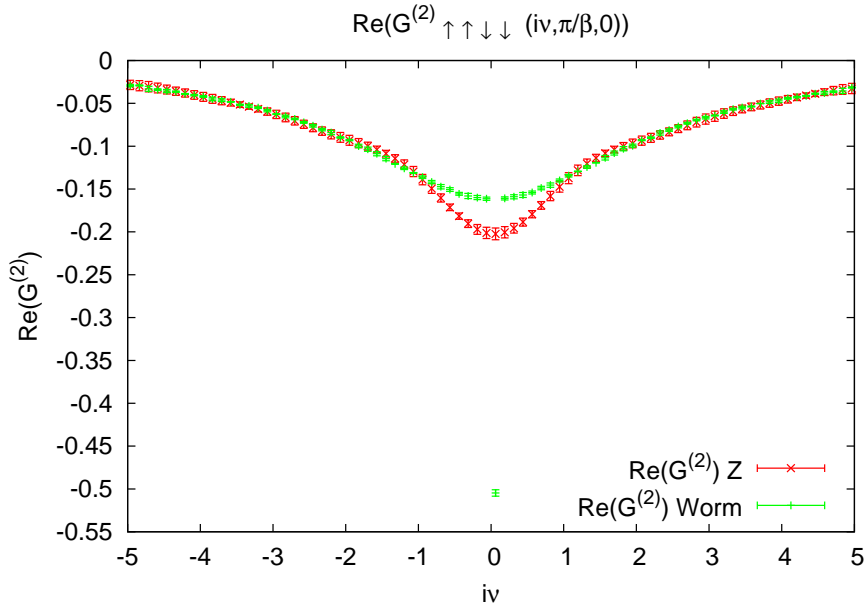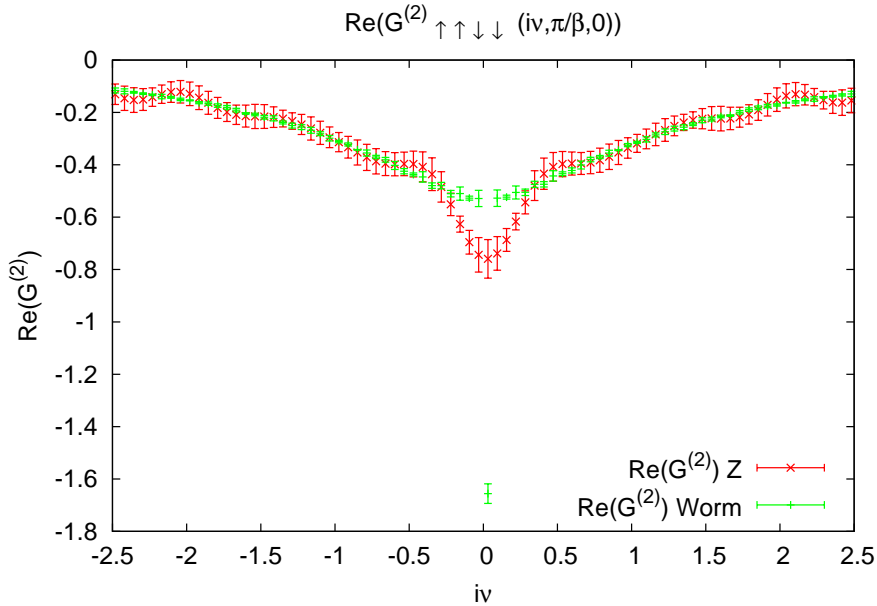

Figure 39: $\Re(G^{(2)}_{\uparrow\uparrow\uparrow\uparrow}(i\nu, i\nu' = \frac{\pi}{\beta}, i\omega = 0))$ with errorbars for the single-orbital, metallic case comparing partition function sampling and worm sampling.
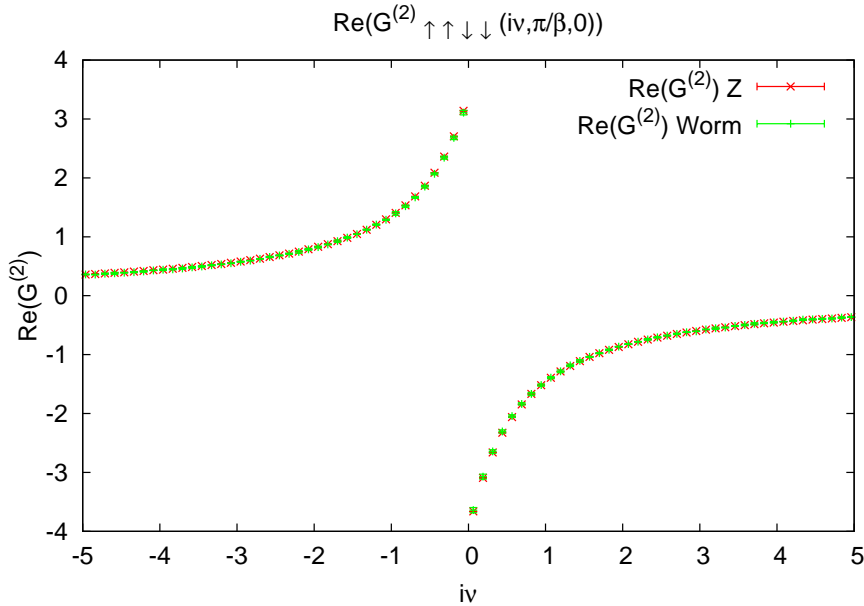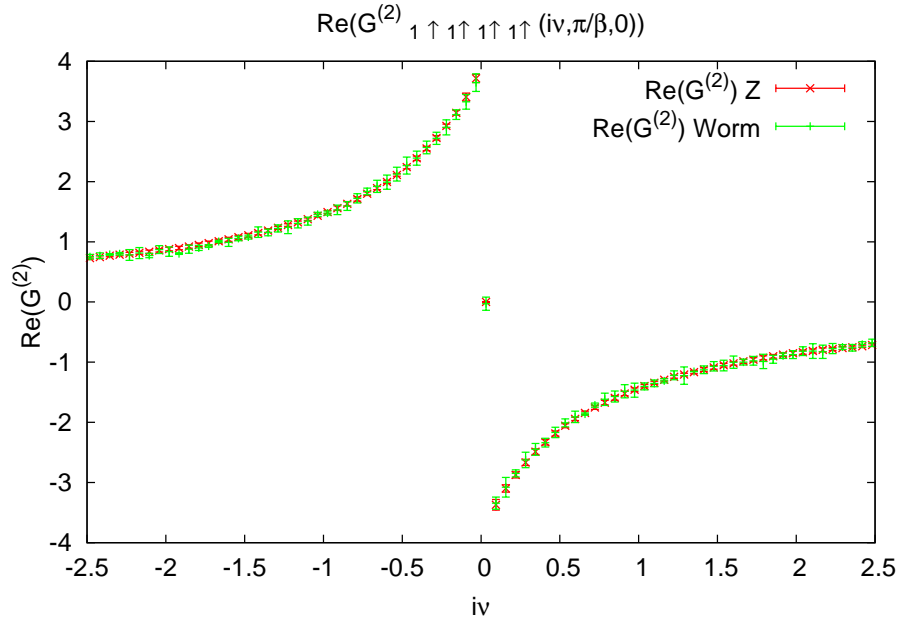
Figure 40: $\Re(G^{(2)}_{\uparrow\uparrow\downarrow\downarrow}(i\nu, i\nu' = \frac{\pi}{\beta}, i\omega = 0))$ with errorbars for the single-orbital, metallic case comparing partition function sampling and worm sampling.

4.2.2 Approaching the Atomic Limit

We now calculate the results of the two-particle Green's function $G^{(2)}$ for the converged DMFT results of the Mott insulating case, at $U = 5.0D$, $\mu = 2.6D$, $\beta = 50/D$. For the worm algorithm we set the probability of worm inserts and removes to 0.1 and the probability of worm replacement moves to 0.2 and $\eta = 4.7 \cdot 10^{-2}/\beta$. While the $\Re(G^{(2)}_{\uparrow\uparrow\uparrow\uparrow}(i\nu, \frac{\pi}{\beta}, 0))$ slices of worm sampling and sampling in partition function space in Figure 41 agree very well, we observe large deviations in the $\Re(G^{(2)}_{\uparrow\uparrow\downarrow\downarrow}(i\nu, \frac{\pi}{\beta}, 0))$ slices in Figure 42. We remind the reader that these deviations are not present in the metallic case.

The mean expansion order of this system is $k/2 = 1$. As a consequence, there is on average a single operator-pair in the trace. However, the two-particle Green's function measurement requires at least two operator-pairs. This makes partition function sampling highly inefficient. A first check to verify if the estimator breaks down, is to increase the expansion order and repeat the measurement in the Mott insulating case. We do this by increasing the inverse temperature to $\beta = 100/D$ and assuming a lower Coulomb repulsion $U = 3.0D$, with $\mu = 1.6D$. For the worm algorithm we adjust $\eta = 7 \cdot 10^{-3}/\beta$. The result is shown in Figure 43. We observe that the deviations cannot be explained by a break-down due to the low expansion order. We suggest comparing the $\uparrow\uparrow\downarrow\downarrow$-component to exact diagonalization results.

Figure 41: $\Re(G^{(2)}_{\uparrow\uparrow\uparrow\uparrow}(i\nu, i\nu' = \frac{\pi}{\beta}, i\omega = 0))$ with errorbars for the single-orbital, strong-insulating case comparing partition function sampling and worm sampling at $U = 5.0D$, $\mu = 2.6D$ and $\beta = 50/D$.
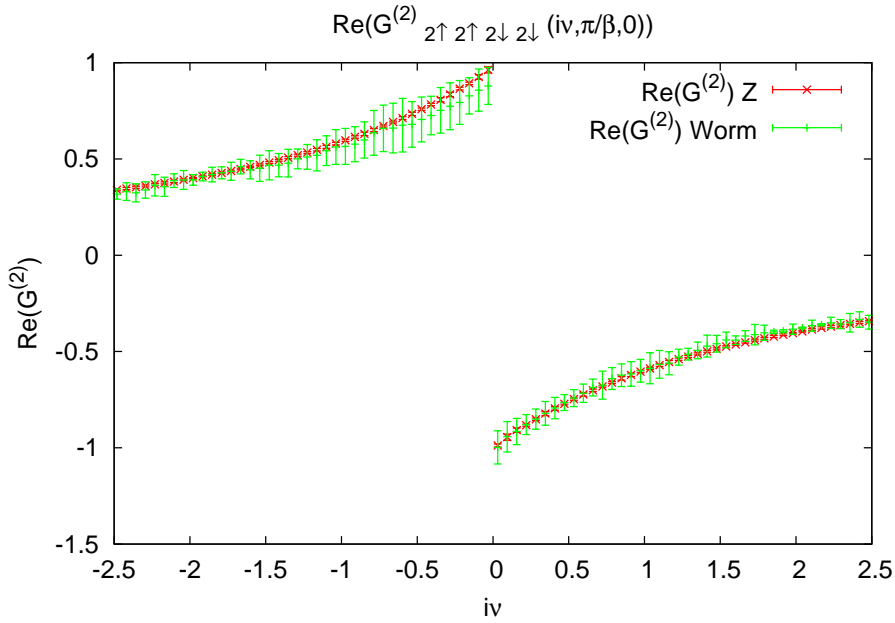

Figure 42: $\Re(G^{(2)}_{\uparrow\uparrow\downarrow\downarrow}(i\nu, i\nu' = \frac{\pi}{\beta}, i\omega = 0))$ with errorbars for the single-orbital, strong-insulating case comparing partition function sampling and worm sampling at $U = 5.0D$, $\mu = 2.6D$ and $\beta = 50/D$.

Figure 43: $\Re(G^{(2)}_{\uparrow\uparrow\downarrow\downarrow}(i\nu, i\nu' = \frac{\pi}{\beta}, i\omega = 0))$ with errorbars for the single-orbital, insulating case comparing partition function sampling and worm sampling at $U = 3.0D$, $\mu = 1.6D$ and $\beta = 100/D$.

4.2.3 Multi-Orbital Slater-Kanamori Results

In this section we present some results from sampling the Green's function $G^{(2)}$ of a multi-orbital system with Slater-Kanamori interaction parameters. Again, we will calculate a two-orbital Bethe lattice with two different bandwidths. Instead of assuming the same interaction parameters as defined for the one-particle Green's function measurement, we now assume a system, where both bands are metallic. We do this to avoid the deviations between worm sampling and sampling in partition function space identified in the previous section. The parameters are set to $D_1 = 1.0$ for the first orbital and $D_2 = 2.0$ for the second orbital. We choose $U = 1.0D_1$ and $J = 0.25D_1$, and $U' = U - 2J = 0.5D_1$. The inverse temperature is set to $\beta = 100/D_1$ and the half-filling condition for the Slater-Kanamori case is given by $\mu = \frac{3}{2}U - \frac{5}{2}J$ resulting in $\mu = 0.875D_1$. For the worm algorithm we set the probability of worm inserts and removes to 0.6 and the probability of worm replacement moves to 0.2. We have chosen a significantly higher probability for worm inserts and removes at this point to account for the asymmetry of the two bands. This way we are able to counter higher rejection rates of worm operators for the second band. As a result, we set $\eta = 2 \cdot 10^{-4}/(2\beta)$ (the factor 2 is an attempt to scale η with the number of orbitals).

Figure 44: $\Re(G^{(2)}_{1\uparrow1\uparrow1\uparrow1\uparrow}(i\nu, i\nu' = \frac{\pi}{\beta}, i\omega = 0))$ with errorbars for the two-orbital, Slater-Kanamori case using partition function sampling and worm sampling.
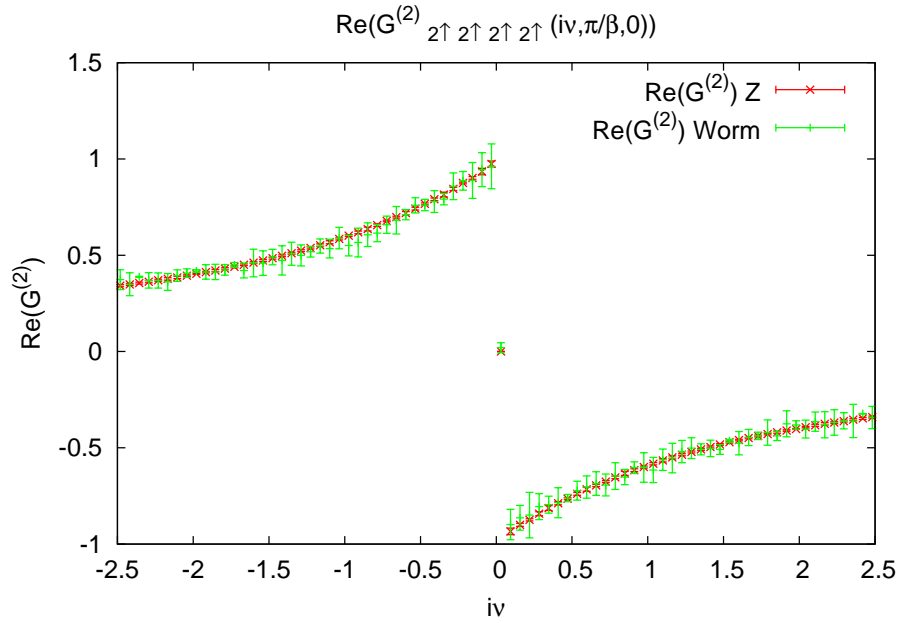
Figure 45: $\Re(G^{(2)}_{2\uparrow2\uparrow2\uparrow2\uparrow}(i\nu, i\nu' = \frac{\pi}{\beta}, i\omega = 0))$ with errorbars for the two-orbital, Slater-Kanamori case comparing partition function sampling and worm sampling.
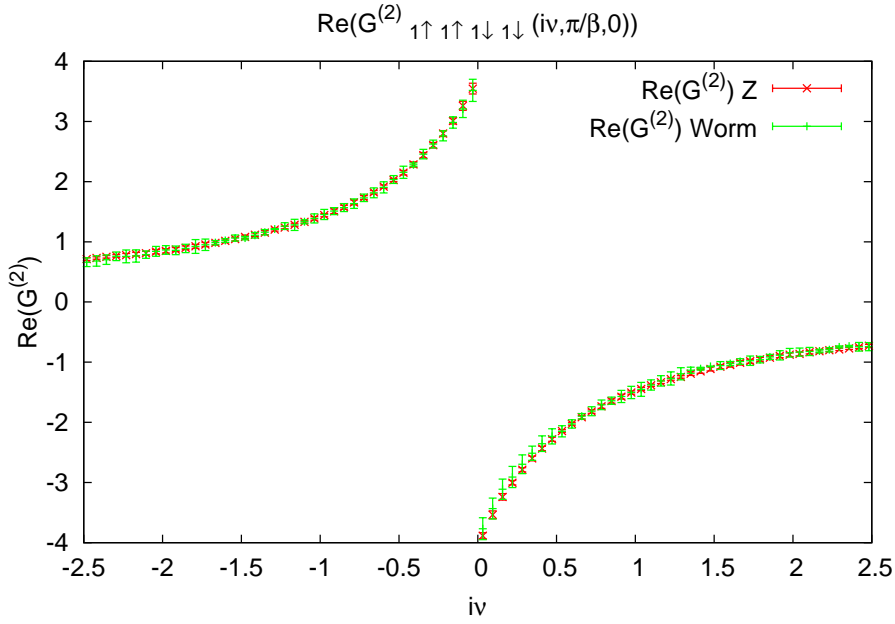
Figure 46: $\Re(G^{(2)}_{1\uparrow1\uparrow1\downarrow1\downarrow}(i\nu, i\nu' = \frac{\pi}{\beta}, i\omega = 0))$ with errorbars for the two-orbital, Slater-Kanamori case comparing partition function sampling and worm sampling.

Figure 47: $\Re(G^{(2)}_{2\uparrow2\uparrow2\downarrow2\downarrow}(i\nu, i\nu' = \frac{\pi}{\beta}, i\omega = 0))$ with errorbars for the two-orbital, Slater-Kanamori case comparing partition function sampling and worm sampling.

79
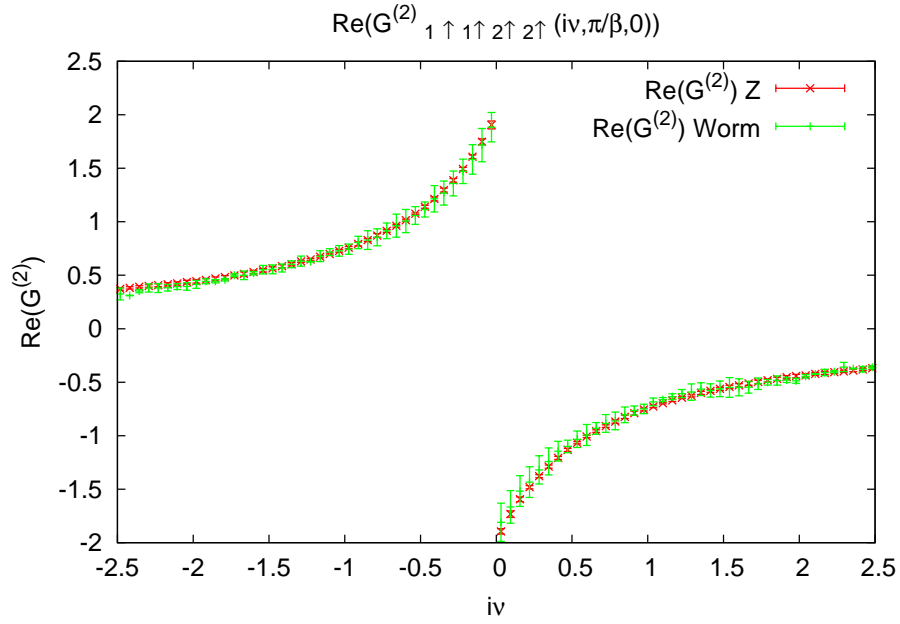
Figure 48: $\Re(G^{(2)}_{1\uparrow1\uparrow2\uparrow2\uparrow}(i\nu, i\nu' = \frac{\pi}{\beta}, i\omega = 0))$ with errorbars for the two-orbital, Slater-Kanamori case comparing partition function sampling and worm sampling.
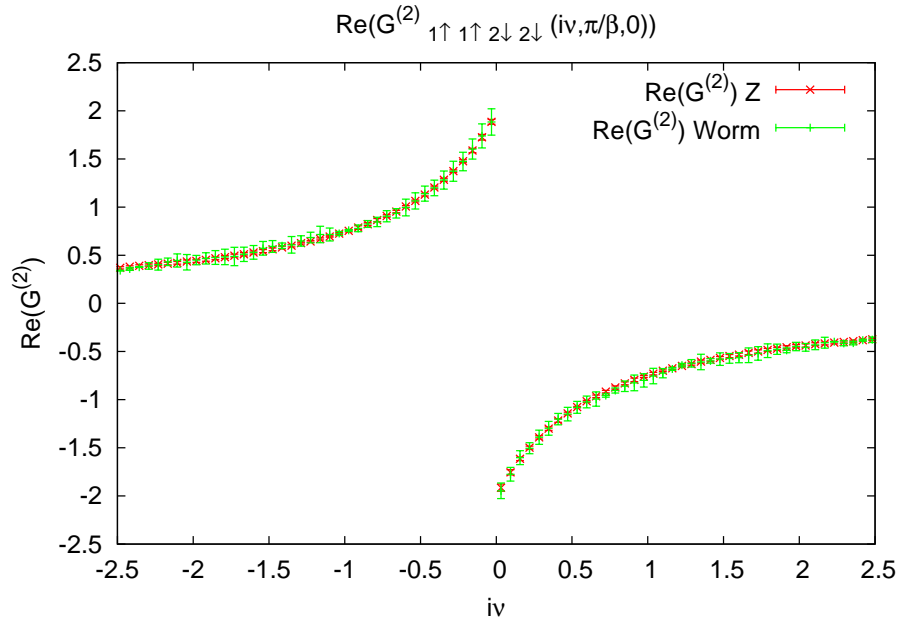
Figure 49: $\Re(G^{(2)}_{1\uparrow1\uparrow2\downarrow2\downarrow}(i\nu, i\nu' = \frac{\pi}{\beta}, i\omega = 0))$ with errorbars for the two-orbital, Slater-Kanamori case comparing partition function sampling and worm sampling.

We observe that worm sampling and sampling in partition function space yield similar results for measuring the two-particle Green's function for the two-orbital, Slater-Kanamori model. In Figure 44 we observe better errorbars in partition function space for the ↑↑↑↑- component. This can be explained by the fact, that the measurement in partition function space implicitly includes the crossing symmetry as we consider all possible combinations of operator-pairs such as $G^2(\tau_1, \tau_2, \tau_3, \tau_4)$ and $G^2(\tau_1, \tau_4, \tau_3, \tau_1)$, while in worm sampling we would need to consider the crossing symmetry explicitly.

Larger errorbars in worm sampling for the slices of the second band in Figures 45 and 47 indicate that it is much more difficult to insert worms into this band than inserting worms into the first band. This asymmetry is due to the fact that we are dealing with two different energy scales D_1 and D_2. We have observed that the factor η, which determines the likelihood of changing into worm space, scales with the interaction parameter U. When dealing with two bands, each having a different interaction parameter U_1 and U_2, we would need to define two parameters η_1 and η_2. This problem is identical to re-weighing the acceptance rates in order to propose more worm inserts into regions of the configuration space, which are less accessible.

5   Conclusion and Outlook

In this work we focused on numerical methods to find solutions to the Hubbard model in order to model strongly correlated materials. In DMFT the Hubbard model is mapped onto the Anderson impurity model, which is solved self-consistently. In this work, we have presented the derivation of the hybridization expansion of the Anderson impurity model in order to solve the quantum impurity problem numerically. We further reviewed the basics of Monte Carlo integration, including the concepts of importance sampling, Markov chains, ergodicity and detailed balance, which lead to the formulation of the Metropolis-Hastings algorithm. We combined the hybridization expansion and the Metropolis-Hastings algorithm to set up the CT-Hyb algorithm. We put a strong emphasis on deriving all contributions to the fermionic sign, as a correct treatment is crucial for extending the CT-Hyb algorithm to include worm sampling. With this foundation, we then derived and implemented the worm algorithm as an alternative sampling scheme in the hybridization expansion. While for partition function sampling in CT-Hyb, we cut hybridization lines to measure Green's functions, in worm sampling we insert local operators explicitly to measure Green's functions. We motivated worm sampling by its advantages in sampling strongly insulating systems and the possibility to sample diagrammatic contributions to the two particle Green's function, which are not part of the partition function sampling (i.e. off-diagonal contributions).

We suggest as a first test to verify that off-diagonal contributions vanish for $G^{(1)}$ and in general for density-density cases. Once it is possible to measure off-diagonal contributions of $G^{(2)}$ for general interactions, the next step is to derive generalized improved estimators, similar to [43], to extract the fully irreducible vertex Λ from the QMC code.

We further showed how measuring the Green's function becomes trivial in the configuration space of worm sampling $\mathcal{C}_{G^{(n)}}$ and how to carry out the normalization procedure. In this work we have normalized Green's functions of worm sampling by repeatedly jumping back and forth between partition function space $\mathcal{C}_Z$ and Green's function space $\mathcal{C}_{G^{(n)}}$. We have further suggested to sample the Green's function exclusively in Green's function space and to find the normalization differently. This way, it would be possible to avoid the re-weighing parameter η. In any case, we suggest to find approximations of the scaling behavior of η in the future. A major advantage of the worm algorithm over partition function sampling is that we have complete control over the worm operators. This way we can easily measure only certain parts of $G^{(n)}$ or re-weigh acceptance probabilities to access configurations suppressed by a vanishing hybridization function. This was already suggested in [26] and is closely related to Wang-Landau sampling.

We have verified that sampling the one-particle Green's function $G^{(1)}$ and the two-particle Green's function $G^{(2)}$ in Green's function space $\mathcal{C}_{G^{(n)}}$ give results comparable to sampling in partition function space $\mathcal{C}_Z$. When sampling $G^{(1)}$ for the single-orbital case, the two algorithms behave very similar for the metallic phase, the insulating phase and the Mott metal-insulator transition. We observe how worm sampling in combination with DMFT results in a better convergence of the self-consistency loop. When sampling $G^{(2)}$ for strongly insulating systems, worm sampling gives different results than the measurement in partition function space. We suggest to investigate these differences in greater detail by comparing our results to exact diagonalization measurements of $G^{(2)}$ for the Mott insulator.

For measuring $G^{(1)}$ in the strongly insulating case, we postpone the analysis of auto-correlation lengths in Green's function space $\mathcal{C}_{G^{(1)}}$ to the future. A relatively simple way to decrease auto-correlation lengths in worm space is to further implement proper shift moves. We have shown that the acceptance rate for worm replacement moves is fully determined by the determinant ratio, while the acceptance rate for worm shift moves is fully determined by the ratio of the weight of the local traces. Finally, we suggest to improve the estimator of worm sampling by including a remove-shift (i.e. sliding) measurement, as already suggested in [42] for operators in partition function space.

While this work is attempting to create a solid foundation for implementing worm algorithms into QMC codes, we emphasize how important future developments of theory and algorithms with respect to worm algorithms in QMC are.
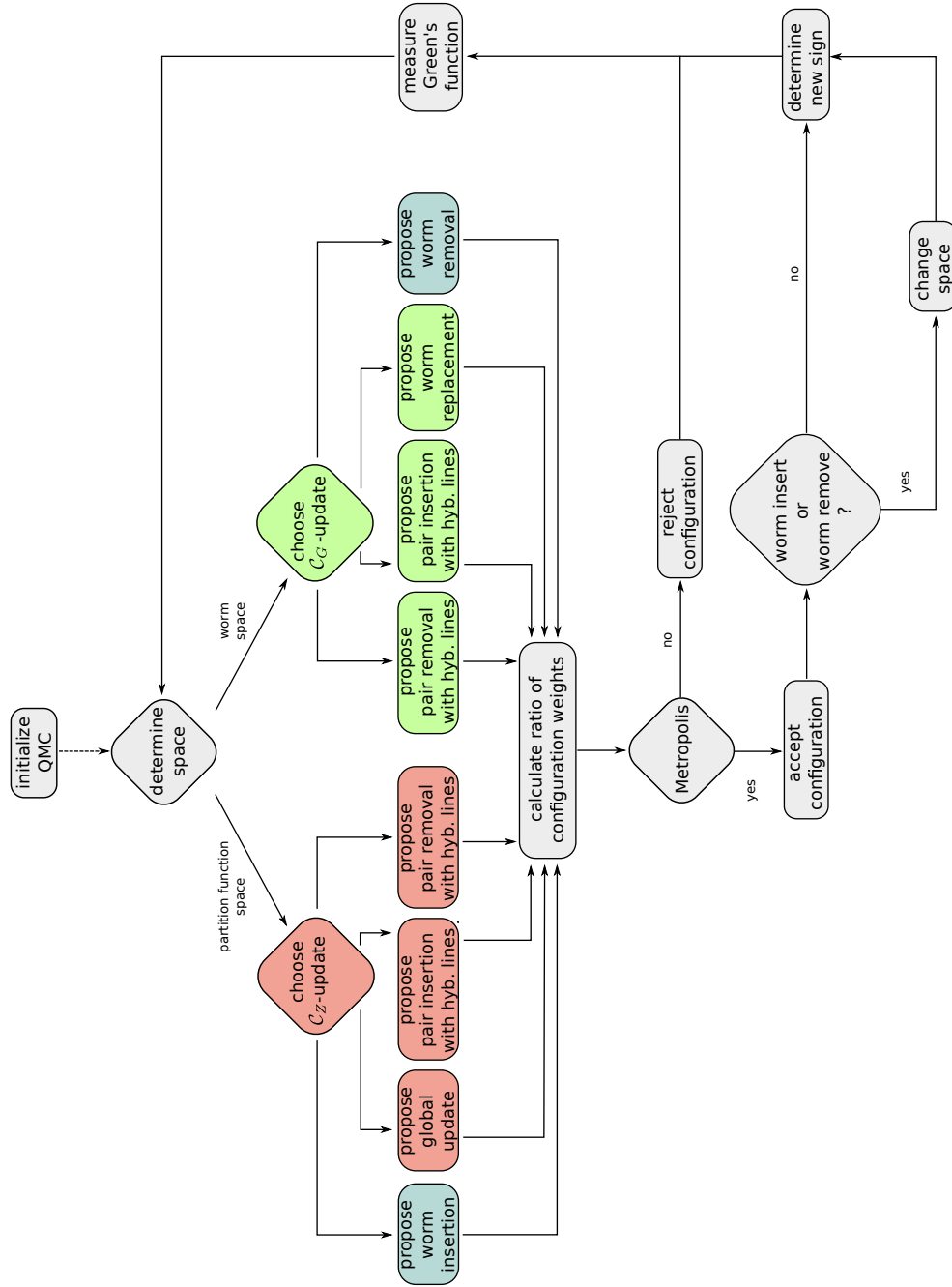
Figure 50: Flow diagram of the worm algorithm implementation for CT-Hyb. The different types of moves are highlighted in color. Moves highlighted in red belong to partition function sampling, moves highlighted in green belong to worm sampling. The worm insertion and worm removal moves cannot be assigned to any of these two categories and are thus marked in blue.

6 References

[1] D. M. Richard. *A Guide to Feynman Diagrams in the Many-Body Problem (Dover Books on Physics)*. Dover Publications, 1992.

[2] W. Kohn and L. J. Sham. Self-consistent equations including exchange and correlation effects. *Phys. Rev.*, 140:1133–1138, Nov 1965.

[3] P. Hohenberg and W. Kohn. Inhomogeneous electron gas. *Phys. Rev.*, 136:B864–B871, Nov 1964.

[4] A. Georges, G. Kotliar, W. Krauth, and M. J. Rozenberg. Dynamical mean-field theory of strongly correlated fermion systems and the limit of infinite dimensions. *Rev. Mod. Phys.*, 68:13–125, Jan 1996.

[5] J. Hubbard. Electron correlations in narrow energy bands. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 276(1365):238–257, 1963.

[6] N. F. Mott. Metal-insulator transition. *Rev. Mod. Phys.*, 40:677–683, Oct 1968.

[7] W. Metzner and D. Vollhardt. Correlated lattice fermions in d=∞ dimensions. *Phys. Rev. Lett.*, 62:324–327, Jan 1989.

[8] M. H. Hettler, A. N. Tahvildar-Zadeh, M. Jarrell, T. Pruschke, and H. R. Krishnamurthy. Nonlocal dynamical correlations of strongly interacting electron systems. *Phys. Rev. B*, 58:7475–7479, Sep 1998.

[9] M. Jarrell, T. Maier, C. Huscroft, and S. Moukouri. Quantum monte carlo algorithm for nonlocal corrections to the dynamical mean-field approximation. *Phys. Rev. B*, 64:195130, Oct 2001.

[10] A. Toschi, A. A. Katanin, and K. Held. Dynamical vertex approximation: A step beyond dynamical mean-field theory. *Phys. Rev. B*, 75:045118, Jan 2007.

[11] A. N. Rubtsov, M. I. Katsnelson, and A. I. Lichtenstein. Dual fermion approach to nonlocal correlations in the hubbard model. *Phys. Rev. B*, 77:033101, Jan 2008.

[12] G. Rohringer, A. Toschi, H. Hafermann, K. Held, V. I. Anisimov, and A. A. Katanin. One-particle irreducible functional approach: A route to diagrammatic extensions of the dynamical mean-field theory. *Phys. Rev. B*, 88:115112, Sep 2013.

[13] P. W. Anderson. Localized magnetic states in metals. *Phys. Rev.*, 124:41–53, Oct 1961.

[14] A. Georges, L. Medici, and J. Mravlje. Strong correlations from hunds coupling. *Annual Review of Condensed Matter Physics*, 4(1):137–178, 2013.

[15] J. Kanamori. Electron correlation and ferromagnetism of transition metals. *Progress of Theoretical Physics*, 30(3):275–289, 1963.

[16] N. Parragh, A. Toschi, K. Held, and G. Sangiovanni. Conserved quantities of $su(2)$-invariant interactions for correlated fermions and the advantages for quantum monte carlo simulations. *Phys. Rev. B*, 86:155158, Oct 2012.

[17] G. Kotliar, S. Y. Savrasov, G. Pálsson, and G. Biroli. Cellular dynamical mean field approach to strongly correlated systems. *Phys. Rev. Lett.*, 87:186401, Oct 2001.

[18] Y. Nomura, S. Sakai, and R. Arita. Multiorbital cluster dynamical mean-field theory with an improved continuous-time quantum monte carlo algorithm. *Phys. Rev. B*, 89:195146, May 2014.

[19] H. Kajueter and G. Kotliar. New iterative perturbation scheme for lattice models with arbitrary filling. *Phys. Rev. Lett.*, 77:131–134, Jul 1996.

[20] H. Keiter and J. C. Kimball. Diagrammatic approach to the anderson model for dilute alloys. *Journal of Applied Physics*, 42(4):1460–1461, 1971.

[21] F. Verstraete, V. Murg, and J.I. Cirac. Matrix product states, projected entangled pair states, and variational renormalization group methods for quantum spin systems. *Advances in Physics*, 57(2):143–224, 2008.

[22] K. G. Wilson. The renormalization group: Critical phenomena and the kondo problem. *Rev. Mod. Phys.*, 47:773–840, Oct 1975.

[23] S. R. White. Density matrix formulation for quantum renormalization groups. *Phys. Rev. Lett.*, 69:2863–2866, Nov 1992.

[24] J. E. Gubernatis, M. Jarrell, R. N. Silver, and D. S. Sivia. Quantum monte carlo simulations and maximum entropy: Dynamics from imaginary-time data. *Phys. Rev. B*, 44:6011–6029, Sep 1991.

[25] P. Werner, T. Oka, and A. J. Millis. Diagrammatic monte carlo simulation of nonequilibrium systems. *Phys. Rev. B*, 79:035320, Jan 2009.

[26] E. Gull, A. J. Millis, A. I. Lichtenstein, A. N. Rubtsov, M. Troyer, and P. Werner. Continuous-time monte carlo methods for quantum impurity models. *Reviews of Modern Physics*, 83(2):349–404, May 2011.

[27] P. Werner and A. J. Millis. Hybridization expansion impurity solver: General formulation and application to kondo lattice and two-orbital models. *Phys. Rev. B*, 74:155107, Oct 2006.

[28] G. Rohringer, A. Valli, and A. Toschi. Local electronic correlation at the two-particle level. *Phys. Rev. B*, 86:125114, Sep 2012.

[29] L. Boehnke, H. Hafermann, M. Ferrero, F. Lechermann, and O. Parcollet. Orthogonal polynomial representation of imaginary-time green's functions. *Phys. Rev. B*, 84:075145, Aug 2011.

[30] N. Metropolis and S. Ulam. The monte carlo method. *Journal of the American Statistical Association*, 44(247):pp. 335–341, 1949.

[31] D. P. Landau and K. Binder. *A Guide to Monte Carlo Simulations in Statistical Physics -*. Cambridge University Press, Cambridge, 3rd edition, 2009.

[32] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes 3rd Edition - The Art of Scientific Computing*. Cambridge University Press, Cambridge, 3rd edition, 2007.

[33] P. Bremaud. *Markov chains: Gibbs fields, Monte Carlo simulation and queues*. Texts in applied mathematics. Springer, New York, Berlin, Heidelberg, 1999.

[34] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1092, 1953.

[35] W. K. Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

[36] J. E. Hirsch and R. M. Fye. Monte carlo method for magnetic impurities in metals. *Phys. Rev. Lett.*, 56:2521–2524, Jun 1986.

[37] P. Werner, A. Comanac, L. de' Medici, M. Troyer, and A. J. Millis. Continuous-time solver for quantum impurity models. *Phys. Rev. Lett.*, 97:076405, Aug 2006.

[38] J. Yoo, S. Chandrasekharan, R. K. Kaul, D. Ullmo, and H. U. Baranger. On the sign problem in the hirschfye algorithm for impurity problems. *Journal of Physics A: Mathematical and General*, 38(48):10307, 2005.

[39] N. V. Prokof'ev, B.V. Svistunov, and I.S Tupitsyn. "worm" algorithm in quantum monte carlo simulations. *Physics Letters A*, 238(4-5):253–257, Feb 1998.

[40] N. V. Prokof'ev, B. V. Svistunov, and I. S. Tupitsyn. Exact, complete, and universal continuous-time worldline monte carlo approach to the statistics of discrete quantum systems. *Journal of Experimental and Theoretical Physics*, 87(2):310–321, 1998.

[41] E. Gull, D. R. Reichman, and A. J. Millis. Bold-line diagrammatic monte carlo method: General formulation and application to expansion around the noncrossing approximation. *Phys. Rev. B*, 82:075109, Aug 2010.

[42] P. Augustinsky and J. Kunes. Improved greens function measurement for hybridization expansion quantum monte carlo. *Computer Physics Communications*, 184(9):2119 – 2126, 2013.

[43] H. Hafermann, K. R. Patton, and P. Werner. Improved estimators for the self-energy and vertex function in hybridization-expansion continuous-time quantum monte carlo simulations. *Phys. Rev. B*, 85:205106, May 2012.

Index

Acknowledgements