Efficient non-equidistant FFT approach to the measurement of single- and two-particle quantities in continuous time Quantum Monte Carlo methods

Peter Staar\textsuperscript{1}, Thomas A. Maier\textsuperscript{2}, and Thomas C. Schulthess\textsuperscript{1,2}

\textsuperscript{1} Institute for Theoretical Physics, Wolfgang Pauli Strasse 27, ETH Zurich, 8093 Zurich, Switzerland
\textsuperscript{2} Computer Science and Mathematics Division and Center for Nanophase Materials Sciences, Oak Ridge National Laboratory, Oak Ridge, TN 37831

Abstract. Continuous time cluster solvers allow us to measure single- and two-particle Greens functions in the Matsubara frequency domain with unprecedented accuracy. Currently, the usage of the two-particle functions is limited due to a lack of an efficient measurement method that can deal with the random times of the vertices. In this paper, we show how the Non-equidistant Fast Fourier Transform (NFFT) algorithm can be modified in order to obtain a very efficient measurement algorithm. For the single particle case, we propose a delayed-NFFT (d-NFFT) scheme, which reduces the arithmetical operations from $O(N \log(N))$ in NFFT to $O(N)$, a huge improvement compared to the standard $O(N^2)$ of the Non-equidistant Discrete Fourier Transform (NDFT), currently used in most continuous time cluster solvers. For the two-particle case, we discuss how the NFFT can be applied to measure the two-particle Greens functions and how to exploit its properties to further optimize the NFFT. We then apply these algorithms to the half-filled 2D Hubbard model at $U/t = 8$ in order to study the anti-ferromagnetic transition. In particular, we confirm the logarithmic decay of the Neel-temperatures versus cluster-sizes in accordance with the Mermin-Wagner theorem.

1. Introduction

The classification of different phases in Fermionic lattice models with strong repulsive interactions is one of the fundamental questions in solid state theory. Many methods have been developed to address this question, each with its own strengths and weaknesses. Exact diagonalization is adequate to investigate these models for small systems, but cannot be used to study the thermodynamic limit, and is thus not usable for the study of, e.g. critical temperature in phase transitions. The Dynamical Matrix Renormalization Group can analyze larger system sizes, but going beyond quasi 1 dimensions has proven difficult. To bridge the gap between dimensionality and thermodynamics, the Dynamical Cluster Approximation \cite{1} was developed over the last ten years. Used in combination with a Quantum Monte Carlo (QMC) cluster-method, it has become the approach of choice to investigate phase transition with non-local order-parameters \cite{2}. Although it is plagued by the fermionic sign problem at low temperatures, it has proven to predict phase transitions such as d-wave superconductivity in the 2D Hubbard model \cite{3} and anti-ferromagnetism in the 3D Hubbard model \cite{4}.

Most of the results produced with the DCA have been obtained with a Hirsch-Fye QMC cluster method \cite{5}. The Hirsch-Fye algorithm is based on a Trotter-decomposition of the
Hamiltonian and is thus only exact in the limit of an infinite number of time-slices. Therefore, each physical quantity obtained with a Hirsch-Fye cluster method needs to be carefully extrapolated towards zero time-step size \[4\]. Recently, a new generation of continuous time cluster-methods has been introduced \[6\], which overcome this problem. These methods are more accurate, and make extrapolations to zero time-step unnecessary.

The continuous time methods do not suffer from the discretization problem of the Hirsch-Fye algorithm because the vertices are allowed to reside on randomly chosen times. This extra freedom leads to a better sampling of the partition function, but it also introduces a new set of problems when measuring physical quantities. In continuous time Monte Carlo methods, quantities are measured in either imaginary time or in Matsubara frequencies. Measuring in imaginary time is very fast but it introduces errors via the binning, since the locations along the time axis for which observables are being measured are changing between measurements. Measuring in Matsubara frequencies on the other hand is exact, but slow, since each measurement requires a discrete Fourier transform. Efficient algorithms such as FFT can not be applied, since they require data on an equispaced time-grid. The measurement of the Greens functions and other related functions are therefore generally slow for continuous time methods, especially in the case of the two-particle Greens functions. This inefficiency limits drastically the usage of the two-particle Greens functions to detect phase-transitions, and hence the DCA as such.

In the implementation section of this paper, we will present methods to accelerate the measurements of single- and two-particle functions in the context of continuous time cluster methods. The focus in this paper lies on the CT-AUX cluster algorithm \[7\] \[8\], but the findings can be applied to other continuous time methods without complications. For the single particle Greens function, we optimize the existing algorithm Non-equispaced Fast Fourier Transform (NFFT) \[9\] for a Monte Carlo integration such that measurements scale linear in the expansion order \(\langle k \rangle\), and not as \(\sim O((\langle k \rangle \log(\langle k \rangle)))\) of the NFFT. For the two-particle Greens function, we discuss various methods that exploit the properties of the two particle Greens functions to accelerate the NFFT on these two-dimensional domains.

Due to the improvements on the measurements, we were able to conduct a systematic study over a wide range of cluster sizes of the half-filled 2D single-band Hubbard model. This model is defined by its Hamiltonian

\[
H = -t \sum_{\langle i,j \rangle, \sigma} c_{i,\sigma}^\dagger c_{j,\sigma} + U \sum_i (n_{i,\uparrow} - \frac{1}{2})(n_{i,\downarrow} - \frac{1}{2}),
\]  

where, \(c_{i,\sigma}^\dagger\) destroys (creates) an electron with spin \(\sigma\) on site \(i\), \(n_{i,\sigma}\) corresponds to the number operator, \(t\) is the hopping amplitude between nearest neighbors \(i\) and \(j\), and \(U\) the on-site Coulomb repulsion. At half filling, the QMC sign problem is absent, which allows us to investigate very large clusters with large interaction strength \(U\). The large cluster runs show the effectiveness of our proposed methods. Here we are interested in the antiferromagnetic behavior of this model. The Mermin-Wagner theorem states that magnetic order cannot exist at finite temperatures in the 2D model. The DCA method, however, because of its mean-field character for finite cluster size, predicts magnetic order at finite temperatures. Here we show that the antiferromagnetic transition temperature \(T_N\) decreases logarithmically with increasing cluster size and give a heuristic argument to explain this behavior.

The paper is structured as follows: In Sec.\[2\] we will briefly review the use of the two-particle Greens functions in the context of the DCA to determine the phase-transitions. In Sec.\[3\] we will first briefly discuss the role of the new methods within the implementation of the DCA. Next, the d-NFFT-measuring method for single-particle quantities is explained in detail. Both accuracy and speed-up of the d-NFFT are compared to the exact NDFT method. For the two-particle
measurements, we will first discuss how the 2D-NFFT can be used, and then discuss various optimization techniques to improve the speed-up. The optimization techniques will be motivated both from the physics and from the algorithm viewpoint. Sec. 5 contains the summary.

2. Theory

The ability to measure non-local two-particle Greens-functions is an important advantage of the DCA towards other methods which study correlated electron systems. With the two-particle Greens-functions, we can solve the Bethe-Salpeter (BS) equation, and accurately determine the critical temperature of a phase transitions. Moreover, the solutions of the BS equation will reveal the internal momentum and frequency structure of the process, which drives the phase-transition. Currently, the use of the two-particle Greens function in a DCA with continuous time cluster solvers is limited due to its time-consuming nature. Up till now, it has therefore mainly been used in fixed time algorithms \[5\], where the two-particle measurements can be efficiently implemented \[10\]. Here, we revisit briefly how the single and two-particle Greens function are defined in a continuous-time algorithm.

In the DCA, the infinite lattice-problem is reduced to a finite size quantum cluster impurity with periodic boundary conditions, embedded into a self-consistent mean-field. This reduction is achieved via a coarsegraining procedure of the Greens functions, in which the Brillouin zone is divided into \(N_c\) patches and the self-energy \(\Sigma\) is assumed to be constant on these patches. In this way, all correlations within the cluster can be dealt with exactly, while long-range correlations outside the cluster are described via a mean-field. In this picture, the single-particle Greens-function \(G_I\) and two-particle Greens-function \(G_{II}\) are translation invariant and therefore defined as

\[
G_I^\sigma[k] = \frac{1}{N_c} \sum_{i,j} e^{i \mathbf{k} \cdot (\mathbf{r}_j - \mathbf{r}_i)} \langle T_\tau \{ \hat{c}_{\mathbf{r}_j,\sigma} \hat{c}_{\mathbf{r}_i,\sigma} \} \rangle,
\]

\[
G_{II}^{\sigma_1,\sigma_2}[k_1, k_2, q] = \frac{1}{N_c} \sum_{r_1, r_2, r_3, r_4} e^{i (k_1 + q) \cdot r_1} e^{-i k_1 \cdot r_2} \times e^{i k_2 \cdot r_3} e^{-i (k_2 + q) \cdot r_4} \langle T_\tau \{ \hat{c}_{\mathbf{r}_1,\sigma_1} \hat{c}_{\mathbf{r}_2,\sigma_1} \hat{c}_{\mathbf{r}_3,\sigma_2} \hat{c}_{\mathbf{r}_4,\sigma_2} \} \rangle,
\]

where we have used the conventional notation \(k = \{ \mathbf{k}, \varpi \}\) and \(r = \{ \mathbf{r}, \tau \}\ \[11\]. The susceptibility \(\chi\) is now computed from the single and two-particle Greens function in a two step process. First, one obtains the cluster vertex function \(\Gamma_c\) via inversion of the Bethe-Salpeter equation on the cluster

\[
\Gamma_c = \left[ G_{II}^0 \right]^{-1} - \left[ G_{II} \right]^{-1}.
\]

Here, \(G_{II}^0\) represents the noninteracting cluster-susceptibility, which is obtained from a pair of interacting Greens functions. The lattice susceptibility is now computed by inverting a second time the Bethe-Salpeter equation, which is defined on the infinite lattice and uses the \(\Gamma_c\) as an approximation for the vertex function. The noninteracting lattice-susceptibility \(\chi_0\) corresponds now to a coarsegrained pair of interacting Greens-functions, and the lattice susceptibility is computed as

\[
\chi = \chi_0 \left[ 1 - \Gamma_c \chi_0 \right]^{-1}.
\]

Since a phase transition occurs at the divergence of \(\chi\), it is sufficient to look at the leading eigenvalue \(\lambda_0\) of the matrix \(\Gamma_c \chi_0\), and observe that it crosses 1 as the temperature is lowered. In this paper, we will consider the anti-ferromagnetic transition, in which case we compute
the two-particle Greens function that corresponds to the magnetic particle-hole channel. Its corresponding two-particle Greens function \(G^{(m),II}\) and noninteracting lattice-susceptibility \(\chi_0^{(m)}\) are defined respectively as,

\[
G^{(m),II}[k_1, k_2, q] = \frac{1}{4} \sum_{\sigma_1, \sigma_2 = \pm 1} (\sigma_1 \sigma_2) G^{II}_{\sigma_1, \sigma_2}[k_1, k_2, q],
\]

\[
\chi_0^{(m)}[k_1, k_2, q] = \frac{\delta_{k_1, k_2}}{N_c} \int_{k \in k_1} dk G(k) G(k + q).
\]

3. Implementation.

The DCA algorithm is schematically represented in Fig. 1. For a good implementation, it is crucial to have a fast QMC-cluster solver, since other sections only take a marginal amount of the run-time. All continuous time QMC-solvers consists of 2 parts that are executed consecutively. The non-invariant single particle Greens functions are only used during the computation of the two-particle case. We present here methods based on NFFT to accelerate the measurements of Greens functions in the context of continuous time cluster solvers. The focus in this paper lies on the CT-AUX cluster solver, but the findings can be applied to other continuous time solvers without complications, e.g. such as the Hybridization-solver.

3.1. measurement of single particle quantities

There exist two types of single particle quantities. The most common ones, such as the self-energy and the single-particle Greens function \(G^I\) are translation invariant in space and time. The non-invariant single particle Greens functions are only used during the computation of the two-particle Greens function. We focus first on the space-time translation invariant quantities. In the CT-AUX algorithm, the single-particle Greens function \(G^I\) is not directly obtained via the Monte Carlo integration, but derived from another accumulated quantity \(\langle M[k, i \omega] \rangle\). In the latter, \(\langle ... \rangle\) represents the Monte Carlo integration. Following closely the notation in [7], we can write down the quantity \(M\) as a function of the \(N\)-matrix and its corresponding Hubbard-Stratonovitch spin configuration \(\{v\}\),

\[
G^I[k] = G^0[k] - G^0[k] \langle M[k] \rangle G^0[k],
\]

\[
M^{(v)}[r, i \omega] = \sum_{\lambda_1, \lambda_2 \in \{v\}} e^{i \omega (\tau_{\lambda_2} - \tau_{\lambda_1})} M^{(v)}[\lambda_1, \lambda_2]
\]

\[
M^{(v)}[\lambda_1, \lambda_2] = \delta_{r, \lambda_2 - \lambda_1} (r e^{V\{\lambda_1\}} - 1) N[\lambda_1, \lambda_2].
\]

In the case of the Hybridization algorithm, one directly accumulates the single particle Greens
Compute the self-energy $\Sigma$.

Quantum Monte Carlo integration.

$$G_\sigma(\vec{K}, \omega) = \prod_{i=1,2} \left( \int_0^\beta d\tau_i \sum_{\vec{R}_i} e^{i \vec{R}_i (\tau_2 - \tau_1)} \times \langle c^{\dagger}_\sigma(\vec{R}_1, \tau_1) c_\sigma(\vec{R}_2, \tau_2) \rangle \right)$$

Thermalization of the random walk.

Do (submatrix-updates.)
while(expansion order is not converged)

Stop if the Self-energy is converged,
else continue the loop.

Sampling of the Greens-functions.

Measure the Greens-functions.

- $d$-NFFT: local convolutions.
  $$\mathcal{M}(\vec{R}_i - \vec{R}_i, \tau)$$
  $$+ = \int d\vec{r} \phi(\vec{r} - \vec{r}) \mathcal{M}_{ij} \delta(\vec{r} - (\tau_j - \tau_i))$$

- 2D-NFFT: accumulate $G^{II}$ (last iteration)

Obtain the Greens-function $G$ from $\mathcal{M}(\vec{R}, \tau)$

- $d$-NFFT: FFT and rescaling.
  $$\langle \mathcal{M}(\vec{K}, \omega) \rangle \leftarrow \frac{1}{\phi(\omega)} FFT[\mathcal{M}(\vec{R}, \tau)]$$

- Obtain new Greens function.
  $$G^{I}(\vec{K}, \omega_m) = G^{II}(\vec{K}, \omega_m) \left\{ 1 - \langle \mathcal{M}(\vec{K}, \omega) \rangle G^{II}(\vec{K}, \omega_m) \right\}$$

Figure 1: A schematic representation of the DCA self-consistency loop, in which a CT-AUX cluster solver is used. The sections affected by recent optimizations are highlighted (color online). The thermalization of the random walk and the sampling of the partition function (green) have been accelerated by the submatrix update-procedure $\mathbb{S}$. The measurement of the single and two-particle functions (red) are accelerated respectively by the $d$-NFFT and 2D NFFT procedures, which are discussed in this paper. Notice that the $d$-NFFT algorithm spans over two section of the Monte Carlo integration: In the sampling-phase of the Greens-function, it will convolute the samples of the function $\mathcal{M}(\vec{r}, \tau)$ with a localized window-function $\phi$ to obtain $\mathcal{M}$ and accumulate the latter. After the sampling phase, it will Fourier transform this convoluted function $\mathcal{M}$ and rescale it by the appropriate Fourier coefficient of the window-function, in order to obtain the averaged Fourier transform of $\langle \mathcal{M}(\vec{r}, \tau) \rangle$. The latter is then used to obtain the interacting Greens function $G^{I}$, from which a new Self-Energy $\Sigma$ can be obtained.
function $G^I$ according to Eq. (3) from the inverted Hybridization matrix $M = [F(\tau_j - \tau_j)]^{-1}$,

$$G^I = \sum_{i,j} M_{i,j} \exp^{i \varpi (\tau_j - \tau_i)}.$$  \hspace{1cm} (3)

In both solvers, the measurement boils down to a Fourier transform of non-equispaced data since the vertices reside on randomly chosen imaginary times. This eliminates the use of the fast Fourier transform (FFT) because the latter requires equispaced data. In a straightforward implementation of the non-equispaced discrete Fourier transform (NDFT), each measurement would be a two step process. One would first compute the Fourier coefficient matrix $C_{m,l} = e^{i \varpi_m \tau_l}$, which involves the evaluation of many expensive trigonometric functions. Next, the actual transform would be carried out via a matrix-vector multiplication. Both operations scale as $O(N_{\varpi} N_{\tau})$, where $N_{\varpi}$ and $N_{\tau}$ represent, respectively, the number of positive matsubara frequencies and the number of vertices.

We present here a measurement algorithm which scales linear in $N_{\tau}$, and requires only one Fourier transform at the end of each Monte-Carlo integration. The dependency of our measurement algorithm on $N_{\varpi}$ is therefore virtually eliminated and a speed-up of order $N_{\varpi}$ can be expected. It is instructive to begin with a short review on the non-equispaced fast Fourier transform (NFFT) algorithm [9]. The NFFT algorithm is based on the convolution theorem, shown in Eq. (4). It states that the Fourier transform of a convolution is equal to the point-wise product of the Fourier transforms:

$$f \varpi \phi \varpi = \int_0^\beta d\tau e^{i \varpi \tau} \left[ \int_0^\beta d\lambda \phi(\tau - \lambda)f(\lambda) \right].$$  \hspace{1cm} (4)

The NFFT algorithm makes use of this theorem in the following manner. First, the non-equispaced data-set $\{\tau_l, f_l\}$ is projected onto an equispaced grid, with $m N_{\varpi}$ intervals. Here, $m$ is called the oversampling factor and is typically in the range of 4 to 10. This projection is achieved via a convolution with a localized, translation invariant kernel $\phi$, such as a Gaussian distribution function. The localization of the kernel is in the range of $[-m, m]$ lattice spacings, which insures that the time needed for the convolution scales linearly with the data size. Next, the FFT algorithm is applied on the projected, now equispaced data and one obtains its representation in Matsubara frequencies. At last, each data-point in Matsubara frequencies is renormalized by its corresponding Fourier-coefficient of the kernel $\phi$, and hence we recover the Fourier component $f_{\varpi}$ of the original non-equispaced data-set. The NFFT algorithm can be summarized as,

$$f_{\varpi} \leftarrow \frac{1}{\phi_{\varpi}} \text{FFT}\left\{ f_l = \sum_\lambda \phi(\tau_l - \frac{l \beta}{N}) f(\lambda) \mid l \in 1, ..., m N_{\varpi} \right\} = \text{NFFT}(\{\tau_l, f_l\})$$  \hspace{1cm} (5)

From Eq. (5), it is clear that the NFFT algorithm scales as $O(m N_{\varpi} \log(m N_{\varpi}) + m N_{\tau})$. An important property of the NFFT-algorithm is that for particular kernels the maximum relative error on $f_{\varpi}$ can be proven to decrease exponentially as a function of the oversampling parameters $m$ [9]. In the case of a gaussian window function $\phi_g$, the maximum relative error $E_2$ is

$$E_2 = \frac{|f_{\varpi}^{\text{NFFT}} - f_{\varpi}^{\text{NDFT}}|}{|f_{\varpi}^{\text{NDFT}}|_1} = 4 e^{-m \pi (1-1/(2\sigma-1))}.$$  

In the case of a Monte-Carlo integration with a continuous time solver, the latter will produce many data-sets, all on a different set of randomly chosen times. Since the FFT is a linear operation, we can delay this operation until the end of each Monte-Carlo integration. In this
way, one accumulates the convoluted function $M(r, \tau_i)$ during the Monte-Carlo integration, instead of $M[r, i \omega]$. 

$$M[r, \tau_i] = \sum_{\lambda_1, \lambda_2 \in \{v\}} \phi((\tau_{\lambda_2} - \tau_{\lambda_1}) - \tau_i) M^{(v)}[\lambda_1, \lambda_2] \quad \text{with} \quad \tau_i = \frac{2 i \beta}{m N_{\omega}} \quad (6)$$

As a result, only a single Fourier transform and rescaling has to be performed on $M[r, \tau_i]$ to obtain $\langle M[k, \omega] \rangle$ at the end of each Monte Carlo iteration. From the latter, $G^f$ can be obtained in the usual manner according to Eq. (2). As shown in Fig. [1], this delayed-NFFT (d-NFFT) will now span over two sections in the Monte-Carlo integration. In the accumulating phase, it will convolute the function-samples on the oversampled NFFT-grid. As a consequence, each measurement now boils down to a convolution, and the desired scaling linear $O(m N_r)$ is obtained. Only after the Monte-Carlo integration, we will Fourier transform the oversampled function $M[r, \tau_i]$. Notice that the delay of the FFT will not affect the accuracy, and the relative error will still decrease exponentially as a function of the oversampling. A second optimization can be obtained by replacing the exact evaluation of the kernel $\phi$ with an interpolated value. Evaluating a Gaussian distribution function involves the evaluation of trigonometric functions, which require many cycles to compute. A drastic improvement in the speed-up can be obtained when $\phi$ is interpolated, but generally at a loss of accuracy. The accuracy can be improved via an adequate tuning of the oversampling factor $m$. This is illustrated in Fig. [2], which shows the relative error $E_2$ for different interpolation schemes. As expected from the theory, one can observe an exponential decline if the kernel is evaluated exactly. For linear and cubic interpolated kernels, we observe that the error saturates as a function of $m$, due to systematic errors. For linear interpolation, the errors saturates at $\approx 10^{-4}$, while in the cubic regime it saturates at machine precision. In Fig. [3] we show the speed-up of d-NFFT over the NDFT.

Algorithm 1 The delayed-NFFT algorithm, which computes the single-particle Greens-function $G^f$.

**Require:** N-samples $\{v\} = \{(\tau_1, M(\tau))\}$ of function $M(\tau)$ at random times in the interval $[-\beta, \beta]$.

**Ensure:** Return the average of the Fourier-transformed samples $\langle M(\omega) \rangle$.

Define the function $M(\tau_1)$ on the oversampled grid $\{\tau_1 = -\beta + l_1 \Delta \tau \mid l_1 \in \{0, \ldots, m N_{\omega}\}\}$ and with $\Delta \tau = \frac{2 \beta}{m N_{\omega}}$.

for $i = 1 \rightarrow N$ do

for $\{\tau_1, M(\tau_1)\} \in \{v\}_i$ do

$I = \frac{\tau_1 + \beta}{\Delta \tau}$ \{find the lower-index of $\tau_i$ such that $\tau(I) \leq \tau_1 < \tau(I + 1)$\}

for $\lambda = -m \rightarrow m$ do

$\phi_{\text{interp}} \leftarrow \text{interpolate } \phi \text{ at } \tau(I + \lambda) - \tau_1$

$M(I + \lambda) \leftarrow M(I + \lambda) + \phi_{\text{interp}} * M(\tau_1)$

end for

end for

end for

Compute $M(\omega) \leftarrow FFT[ M(\tau) ]$

Compute $\langle M \rangle \leftarrow \frac{M(\omega)}{N_{\omega} \phi(\omega)}$

Compute $G^f(\omega) \leftarrow G^0(\omega) [1 + \langle M(\omega) \rangle G^0(\omega)]$

end for
Notice that the speed-up is of the order $O(N_w) \sim 10^3$. Considering both the need for accuracy and speed-up, the most optimal choice for QMC cluster solvers seems to be a cubic interpolation scheme at $m=5$. In this way, the error on the Fourier transform will always be two to three orders of magnitude smaller than a typical statistical error of a Monte-Carlo integration. The fully optimized d-NFFT algorithm is summarized in Alg. (1).

\[ G_{\sigma_1,\sigma_2}[k_1,k_2,q] = \left\langle G^{I}_{\sigma_1}[k_1,k_1+q] G^{I}_{\sigma_2}[k_2+q,k_2] - \delta_{\sigma_1,\sigma_2} G^{I}_{\sigma_1}[k_1,k_2] G^{I}_{\sigma_2}[k_2+q,k_1+q] \right\rangle. \] (7)

Here, $\left\langle \ldots \right\rangle$ represents the Monte Carlo Integration. Just as in the single-particle case, the $\langle \text{TNSP} \rangle$ Greens functions $G^{(v)\sigma}[k_1,k_2]$ can be computed straightforwardly from the $N$-matrix and its Hubbard-Stratonovitch configuration $\{v\}$.

\[ G^{(v)\sigma}[k_1,k_2] = \delta_{k_1,k_2} G^0[k_1] - G^0[k_1] M^{(v)\sigma}[k_1,k_2] G^0[k_2], \]
\[ M^{(v)\sigma}[k_1,k_2] = \sum_{\lambda_1,\lambda_2 \in \{v\}_\sigma} e^{i\omega_2 \tau_{\lambda_2} - \omega_1 \tau_{\lambda_1}} e^{i(k_2 r_{\lambda_2} - k_1 r_{\lambda_1})}. \] (8)

3.2. measurement of two-particle Greens-function $G^{II}$

The two-particle Greens-function $G^{II}$ is obtained via a Quantum Monte Carlo Integration of a product of translation-noninvariant single-particle TNSP Greens functions $G^{I}_{\sigma}[k_1,k_2]$.

\begin{figure}[h]
\centering
\includegraphics[width=0.4\textwidth]{figure2}
\caption{The relative error $E_2$ versus the oversampling factor $m$. The error decreases exponentially for increasing $m$, as expected from the theory. When the kernel $\phi$ is approximated by an interpolating value, the error becomes independent of the $m$. In the linear regime, the error saturates at $\approx 10^{-4}$, while in the cubic regime it saturates at machine precision $\approx 10^{-16}$.}
\end{figure}

\begin{figure}[h]
\centering
\includegraphics[width=0.4\textwidth]{figure3}
\caption{The speed-up of d-NFFT compared to DFT, versus the oversampling parameter $m$. Replacing the exact evaluation of the kernel $\phi$ with its interpolated value results clearly in a drastic speed-up. For this figure, we used typical parameters for a Monte Carlo simulation ($N_w=512, N_r=1024$).}
\end{figure}
For this, the NFFT algorithm has to be generalized to two dimensions, which is straightforward. One constructs an fine 2D grid of the size \( mN_\omega \) in both dimensions and convolutes the M-function with a 2D localized kernel. In our case, we simply used the product of two 1D Gaussian distributions. Just as in the 1D case, we apply the FFT on this fine equispaced grid, and renormalize each function-value in the Matsubara representation by the associated Fourier component of the localized kernel.

Algorithm 2 A 2D-NFFT algorithm to compute the TNSP Greens-function in the CT-AUX.

Require: A set of samples \( \{ \nu \} = \{ \{ \tau_1, \tau_2, M_{i,j} \} \} \) of the 2D function \( M(\tau_1, \tau_2) \)-function at random times in the interval \([-\beta, \beta] \times [-\beta, \beta] \).

Ensure: Return the corresponding TNSO Greens-function \( G(\varpi_1, \varpi_2) \).

Define the function \( M(\tau_1, \tau_2) \) on the 2D-oversampled grid \( \{ \{ \tau_1 = -\beta + l_1 \Delta \tau, \tau_2 = -\beta + l_2 \Delta \tau \} \mid l_1, l_2 \in \{0, \ldots, mN_\omega \} \} \) and with \( \Delta \tau = \frac{2\beta}{mN_\omega} \).

for \( \{ \tau_1, \tau_2, M_{i,j} \} \in \{ \nu \} \) do
  \( I \leftarrow \frac{\tau_1}{\Delta \tau} \) \{ find the lower-index of \( \tau_1 \) such that \( \tau(I) \leq \tau(I) < \tau(I+1) \} \)
  \( J \leftarrow \frac{\tau_2}{\Delta \tau} \) \{ find the lower-index of \( \tau_2 \) such that \( \tau(J) \leq \tau_2 < \tau(J+1) \} \)
  for \( \lambda_i = -m \rightarrow m \) do
    \( \phi_i \leftarrow \phi(\tau(I + \lambda_i) - \tau_i) \)
    for \( \lambda_j = -m \rightarrow m \) do
      \( \phi_j \leftarrow \phi(\tau(J + \lambda_j) - \tau_j) \)
      \( M(I + \lambda_i, J + \lambda_j) \leftarrow M(I + \lambda_i, J + \lambda_j) + \phi_i \star \phi_j \star M_{i,j} \)
    end for
  end for
  Compute \( M(\varpi_1, \varpi_2) \leftarrow FFT[M(\tau_1, \tau_2)] \)
  Compute \( G(\varpi_1, \varpi_2) \leftarrow \frac{M(\varpi_1, \varpi_2)}{\delta_{\varpi_1, \varpi_2}G^0(\varpi_1) - G^0(\varpi_1)M(\varpi_1, \varpi_2)G^0(\varpi_2)} \)
end for

In contrast to the 1D case, it is not possible to use a delayed FT scheme here. The TNSP Greens function needs to be computed every single time in order to evaluate Eq. (7). As a consequence, the measurement of a two-particle function is generally determined by the speed at which the FFT can be performed on the oversampled grid. We will therefore devote the rest of this section to tricks, which can speed up the FFT. First, we can take advantage of the fact that the function-values are real. This means that the function values in Matsubara representation will have a conjugate symmetry, and it is thus sufficient to compute only half of the Matsubara frequencies. This should speed up the FFT by a factor of two [13]. Another trick that should be used is pruning. Notice that we are actually only interested in a small subset of Fourier components of order \( O(N_\omega^2) \) instead of all them \( O(m^2 N_\omega^2) \). This essentially implies that we can skip the \((m-1)N_\omega \) Fourier Transforms in the second dimension. The FFT in 2D is therefore \( O(m^2 N_\omega^2 \log(m^2 N_\omega^2)) \), but rather of \( O(m^2 N_\omega^2 \log(m N_\omega) + m N_\omega^2 \log(m N_\omega)) \). The pruning of our FFT leads generally to an additional speed-up of roughly 2. In Fig. (4), we show you the combined speed-up of the NFFT method in 2D versus the NDFT. Due to its better scaling, the NFFT is clearly a good improvement in the case of a large number of vertices and a large number of Matsubara frequencies. The algorithm will therefore be hugely beneficial.
in the case of low temperatures and strong correlation.

4. Study of the anti-Ferromagnetic transition in the 2D repulsive Hubbard model.
We will illustrate the effectiveness of our newly proposed measurement techniques by investigating the spin-spin correlation length \( l_c \) in the undoped 2D single band Hubbard model (SBHM) defined in Eq. (1). In particular we will look at the growth of the spin-spin correlation length as a function of the temperature. In this way, we can better understand the physics of the spin-fluctuations, which have been found to mediate the pairing interaction between electrons which is responsible for strong \( d \)-wave superconducting correlations in the 2D Hubbard model [15].

There exists no straightforward and direct way to obtain the correlation length \( l_c \), as it does not have a strict definition. In previous work [3], an artificial antiferromagnetic transition temperature, the Neel-temperature \( T_N \) of the cluster was used to obtain a measure of the correlation length. The motivation for this strategy is the following: As the temperature is lowered, the correlation length grows and eventually reaches the linear cluster size \( \sqrt{N_c} \). At that length scale, the DCA does not describe fluctuations and therefore predicts an antiferromagnetic transition in the cluster, i.e.

\[
l_c(T_N) = \sqrt{N_c}.
\]  

This transition is artificial, since it contradicts the Mermin-Wagner theorem which demands \( T_N = 0 \).

The method of operation to determine the temperature-dependence of the correlation length \( l_c(T) \) is now clear. We will compute the Neel-temperature \( T_N \) for a set of increasingly larger clusters and look at its decay as a function of the cluster size. By simply inverting the curve, we can obtain the growth of the spin-spin correlation length as the temperature is lowered.

This procedure demands a very effective implementation of the DCA, since many large clusters will have to be solved for increasingly lower Neel-temperatures. Previously, the award winning DCA++ code [10] which implements a Hirsch-Fye solver was able to compute the Neel-temperature up to clusters of \( N_c = 32 \). Due to the efficient measurement techniques introduced in this paper and the submatrix updates, we were able to push the maximum cluster size to \( N_c = 100 \). We show the Neel-temperatures obtained with the CT-AUX in Fig. (5), and compare it to the values extracted from [3].

The decrease of the Neel-temperature \( T_N \) as a function of cluster size is clearly visible. Fig. (5) shows that the Hirsch-Fye captures the physics qualitatively well, but overestimates

![Figure 4: The speed-up of an optimized 2D-NFFT compared to NDFT in 2D for an oversampling factor \( m = 6 \), which corresponds to a relative error of \( E_2 \sim 10^{-6} \). The method is clearly advantageous for large number of Matsubara frequencies \( (N_\omega \gg 1) \) in the strongly correlated region \( \langle k \rangle \gg 1 \).](image-url)
the transition temperatures in general. This discrepancy between the two solvers grows as clusters get larger and temperatures lower, due to the Trotter approximation. Via a controlled extrapolation towards infinite time slices, it is possible to avoid this discrepancy between the Hirsch-Fye and CT-AUX cluster solver. This is illustrated in Fig. (6), where the extrapolated Neel-temperature from a Hirsch-Fye solver [4] is compared to the one obtained with CT-AUX for a 3D $N_c = 18$ site cluster. Due to limited computational resources, this extrapolation was not done for the Neel-temperatures presented in Fig. (5), which explains the quantitative difference in the results.

Since spin-spin correlations develop exponentially in temperature in a 2D Heisenberg model, the strong coupling limit of the half-filled 2D Hubbard model, we expect the correlation length $l_c(T)$ to have an exponential temperature dependence $l_c(T) = b e^{1/T}$. Together with Eq. (9), this means that

$$T_N = \frac{a}{\log(b \sqrt{N_c})}.$$  \hspace{1cm} (10)

In Fig. (5), we show a fit of the Neel-temperatures obtained with the CT-AUX using Eq. (10). The agreement is remarkably good for large clusters ($N_c \geq 32$). This confirms that the DCA does not violate the Mermin-Wagner theorem in its exact limit $N_c \to \infty$ and confirms that the correlation length $l_c$ grows exponentially as a function of inverse temperature in the 2D Hubbard model.

With the solutions from the Bethe-Salpeter equation on the $N_c = 100$ cluster, we can get insight on the nature of the interaction leading to the antiferromagnetic correlations. The momentum and frequency structure of the eigenvector $\phi$, associated with the leading eigenvalue, is shown for $\mathbf{k}$-points along the Fermi-surface in Fig (7). The leading eigenvector $\phi$ has very little momentum structure. The anti-ferromagnetic transition can thus be characterized as a condensation of local particle-hole pairs (local magnetic moments). In addition, the
Figure 7: The eigenfunction $\phi_k(\varpi)$ for different k-points on the Fermi-surface ($N_c = 100$ and $\beta = 6$). The eigenfunction has very little momentum dependence, indicating that the antiferromagnetic order parameter is essentially a local particle-hole pair. Notice also that the eigenfunction has a finite asymptotic value for large frequencies, reflecting the role of the instantaneous Hubbard $U$ in the moment formation.

The eigenfunction has a clear finite asymptotic value. This reflects the instantaneous contribution of the Hubbard $U$ to the effective interaction that leads to the moment formation.

5. Summary

In summary, we have presented methods based on NFFT which significantly speed up the measurements in the case of continuous time QMC cluster solvers. These methods were then applied to analyze the magnetic behavior of the half-filled 2D Hubbard model with interaction $U/t = 8$. The results are consistent with an exponential divergence of the antiferromagnetic spin-spin correlation length in inverse temperature, and therefore compliant with the Mermin-Wagner theorem.

Acknowledgement

Work at Oak Ridge National Laboratory (ORNL) was supported by the Laboratory Directed Research and Development fund and the Center for Nanophase Materials Sciences, which is sponsored by the Division for Scientific User Facilities of the Office of Basic Energy Sciences. Computational resources at the National Center for Computational Sciences (NCCS) of ORNL and the Swiss National Supercomputing Center (CSCS) in Lugano, Switzerland were used for the simulations.

References

[1] M. Jarrell, T. Maier, and C. Huscroft, “Quantum Monte Carlo algorithm for nonlocal corrections to the dynamical mean-field approximation,” Physical Review B, 2001.
[2] T. Maier, M. Jarrell, T. Pruschke, and M. Hettler, “Quantum cluster theories,” Reviews of Modern Physics, vol. 77, no. 3, pp. 1027–1080, 2005.
[3] T. Maier, M. Jarrell, T. Schulthess, P. Kent, and J. White, “Systematic Study of d-Wave Superconductivity in the 2D Repulsive Hubbard Model,” Physical Review Letters, vol. 95, p. 237001, Nov. 2005.
[4] P. Kent, M. Jarrell, T. Maier, and T. Pruschke, “ Efficient calculation of the antiferromagnetic phase diagram of the three-dimensional hubbard model,” Physical Review B, vol. 72, p. 060411, Aug. 2005.
[5] J. E. Hirsch and R. M. Fye, “Monte carlo method for magnetic impurities in metals,” Phys. Rev. Lett., vol. 56, pp. 2521–2524, Jun 1986.
[6] 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,” Rev. Mod. Phys., vol. 83, pp. 349–404, May 2011.
[7] E. Gull, P. Werner, O. Parcollet, and M. Troyer, “Continuous-time auxiliary-field monte carlo for quantum impurity models,” EPL (Europhysics Letters), vol. 82, no. 5, p. 57003 (6pp), 2008.
[8] E. Gull, P. Staar, S. Fuchs, P. Nukala, M. S. Summers, T. Pruschke, T. C. Schulthess, and T. Maier, “Submatrix updates for the continuous-time auxiliary-field algorithm,” Phys. Rev. B, vol. 83, p. 075122, Feb 2011.
[9] J. Keiner, S. Kunis, and D. Potts, “Using NFFT3 - a software library for various nonequispaced fast Fourier transforms,” *ACM Transactions on Mathematical Software*, vol. 36, pp. 19:1–19:30, Aug. 2009.

[10] G. Alvarez, M. S. Summers, D. E. Maxwell, M. Eisenbach, J. S. Meredith, J. M. Larkin, J. Levesque, T. A. Maier, P. R. C. Kent, E. F. D’Azevedo, and T. C. Schulthess, “New algorithm to enable 400+ TFlop/s sustained performance in simulations of disorder effects in high-Tc superconductors,” in *SC ’08: Proceedings of the 2008 ACM/IEEE conference on Supercomputing*, (Piscataway, NJ, USA), pp. 1–10, IEEE Press, 2008.

[11] A. A. Abrikosov, L. P. Gorkov, and I. Y. Dzyaloshinskii, *Quantum field theoretical methods in statistical physics*, *International series of monographs in natural philosophy*. Pergamon Press, Oxford, 1965.

[12] P. Werner, A. Comanac, L. de’ Medici, et al., “Continuous-time solver for quantum impurity models,” *Phys. Rev. Lett.*, vol. 97, no. 7, p. 076405, 2006.

[13] P. Werner and A. J. Millis, “Hybridization expansion impurity solver: General formulation and application to kondo lattice and two-orbital models,” *Phys. Rev. B*, vol. 74, no. 15, p. 155107, 2006.

[14] M. Frigo and S. G. Johnson, “The design and implementation of FFTW3,” *Proceedings of the IEEE*, vol. 93, no. 2, pp. 216–231, 2005. Special issue on “Program Generation, Optimization, and Platform Adaptation”.

[15] T. A. Maier, M. S. Jarrell, and D. J. Scalapino, “Structure of the pairing interaction in the two-dimensional hubbard model,” *Phys. Rev. Lett.*, vol. 96, p. 047005, Feb 2006.