Journal of Namibian Studies, 33 S2(2023): 2398–2408 ISSN: 2197-5523 (online)

Quantum Physics: Illustration of Multidimensional Green Functions Using Julia Language

Lizeth Fernanda Silva Godoy[1], Lara-Sinaluisa, Jorge[2], Cristian Luis Inca Balseca[3], Martínez-Nogales, Juan[4], Paredes-Godoy, Magdalena[5], Joseph David Guerra Chávez[6]

[1]Independent researcher, fercha.silva1@gmail.com

[2]Magíster en Informática Educativa, Máster Universitario en Formación Internacional, Especializada del Profesorado Especialidad Física y Química, Doctor en Física, Docente, Investigador en la Facultad de Administración de Empresas, Carrera Gestión del Transporte, "Escuela Superior Politécnica de Chimborazo (ESPOCH)", Riobamba, Ecuador, j_lara@espoch.edu.ec

[3]Escuela Superior Politécnica de Chimborazo (ESPOCH), Facultad de Informática y Electrónica, Escuela: Tecnologías de la Información, Correo electrónico: cristianl.inca@espoch.edu.ec

[4]Magíster en Ciencias de la Educación Aprendizaje de la Física, Ingeniero Mecánico, Docente Investigador en la Facultad de Mecánica, Carrera Ingeniería Automotriz, "Escuela Superior Politécnica de Chimborazo (ESPOCH)", Riobamba, Ecuador, jumartinez@espoch.edu.ec

[5]Magíster en Ciencias de la Educación Aprendizaje de la Física, Ingeniero Mecánico, Docente Investigador en la Facultad de Ingeniería, Carrera Ingeniería Industrial, "Universidad Nacional de Chimborazo (UNACH)", Riobamba, Ecuador, maparedes@unach.edu.ec

[6]Independent researcher, jphguerra95@gmail.com

Abstract

In this article, the numerical level representation of imaginary time Green's functions based on a low rank decomposition of the discrete Lehmann spectral representation (DLR) using an effective spectral density has been presented with illustrative efficiency. The DLR basis consists of a collection of exponentials chosen by interpolation decomposition to ensure stable and efficient recovery of imaginary time Green functions or Matsubara frequency samples (Kaye et al., 2022). This implementation to fit behavior of a low-temperature spinless free fermionics-electron gas quantum system requires much fewer degrees of freedom for the standard discretizations. The basic functions of DLR are explicit; carefully chosen to ensure a stable and accurate

approach, simplifying standard operations. Importantly, the function can be explicitly transformed to the Matsubara frequency domain or obtained directly by interpolation on a Matsubara frequency grid.

Keywords: Green function, DLR, Matsubara frequency, Quantum system, Electron gas, fermionics.

1. INTRODUCTION

Lately, an evident cooperation between mathematicians and physicists has intensified to levels never seen before, in similarities of interaction equivalent to the time of Einstein. This cooperation and enthusiasm have focused on sustainable contributions to the development of unified theories that reconcile two great physical ideas of the twentieth century, specifically to the topics of studies in relativity and quantum field theory, underlying all observed forces and elementary particles.

Obviously, this contribution shows the study of the part concerning quantum theory, whose benefit in understanding has been related in an essential way with the implementation of the new and emerging mathematical machinery; at this point, it refers to the implementation of the Julia and Python programming languages for the implicit understanding of mathematical and physical aspects. While their separate motivations and powerful technologies remain, they have surprisingly revitalized and transformed each other, so in both realms they can be combined to address profound applications of these programming languages across a wide range of physical theories, from elementary particles to cosmology. (Besard et al., 2017) (Tosta et al., 2019)

In this vein, this article addresses the illustration by means of the programming software Julia lying in evaluating the Green's function of a single-level quantum system full of spinless fermionic particles, assuming that such a system could exchange particles and energy with the environment, so that its equilibrium state is a large canonical set. The following illustration is based on the Green function of a gas with free electrons (Ar. Abanov et al., 2022) (By Tullio, et al., 2019); and finally, the Green function of a free spin Fermi gas in a square lattice is calculated. (Kaye et al., 2022)

Incondensed matter physics, the Matsubara Green function technique is a powerful tool for studying the physics of many bodies in strongly correlated systems. In practical calculations, the data of Green's function is stored in the computer's memory. When the Green function is represented in the (Chikano et al., 2019) Matsubara frequency domain G(iω_n) , it exhibits a power law decay at high frequencies. Therefore, the number of Matsubara frequencies required to represent grows rapidly with decreasing temperature. Especially when the Green

function has other indices such as spin, orbit and wavenumber, the data becomes hugeG(iω_n)G(iω_n)at low temperatures. Therefore, there is a great demand for a compact representation of the imaginary time dependence of Green's function on practical calculations. (Chikano et al., 2019)

In essence, this Green function allows descriptions to be made between two or more events in spacetime. Since the advantage is focused on the use of the specific package in Julia (GreenFunc.jl), which has allowed the definition between three feasible alternatives for the construction of temporary meshes (MeshArray), essential to perform the approach in the implementation of the Green's function of a body.

2. Materials and Methods

This scientific article has been developed using the documentary methodology at the descriptive level on the Green's function of imaginary time based on a low-range decomposition of the discrete Lehmann spectral representation (DLR) using an effective spectral density. Therefore, the methodical bibliographic selection was implemented with the systematic review of articles that develop scientific contributions on the aforementioned line of research, the above with support in scientific publications through the use of meta-analysis in reliable sources, such as: Medline, Scopus, Springer, Elsevier ; assuming as a reference criterion for selection of articles, the relevance in scientific contribution of the last 20 years for publications, through critical extraction and content analysis of the information obtained, in order to illustrate the reliability in the empirical implementation through the implementation of algorithms in Julia.

3. Theoretical aspects

Lehmann discrete presentation (DLR) in the Matsubara frequency domain

A DLR can be transformed to the Matsubara frequency with analytic domain. According to Kaye et al. (2022), is established based on the following equation, such that:

$$k(iv_n, \omega) = \int_0^1 k(\tau, \omega) e^{-iv_n^{\tau}} d\tau = \frac{1}{(\omega + iv_n)} \qquad (1)$$

Con Matsubara frequency points:

$$i\omega_n = \begin{cases} i(2n+1)\pi \ \ para\ funciones\ fermiónicas\ de\ Green \\ i2n\pi \qquad para\ las\ funciones\ bosónicas\ de\ Green \end{cases}$$

Being anexpansion of DLR, the following representation:

$$G(\tau) = \sum_{l=1}^{r} k(\tau, \omega_l)\hat{g}_l \quad (2)$$

Thus, the Matsubara frequency domain is transformed as:

$$G(i\omega_n) = \sum_{l=1}^{r} k(iv_n, \omega_l)\hat{g}_l \quad (3)$$

Transformation between imaginary time and Matsubara frequency domains

The DLR coefficients for the representation of a given Green's function in imaginary time and the Matsubara frequency domains are the same; where it simply takes the Fourier transform of the DLR in imaginary time using explicitly (1) to obtain the DLR in Matsubara frequency and inverts the transform explicitly to go in the opposite direction. Thus, having obtained DLR coefficients for a Green function, the representation can be evaluated in any domain. (Kaye et al., 2022)

4. Illustration through Julia Language

4.1. Illustration of the Green's function of a level

The first step in the illustration must be to define the structure of the data using MeshArray to present the Green function of a quantum system of a singlelevel full of spinless fermionic particles. (Beenakker et al., 2004; Gigena et al., 2020; Tosta et al., 2019) With a priori conjecture that the system could exchange particles and energy with the environment, so that its equilibrium state is a largecanonical conjunct (Ar. Abanov et al., 2022; Gigena, et al., 2021; Wenhu et al., 2017). The Green function of a particle hasa simple form in the Matsubara frequency representation: (Schüler and Pavlyukh, 2018)

$$G(i\omega_n) = \frac{1}{(i\omega_n - E)} \quad (4)$$

Where:E it is the level of energy. Then, for illustrative purposes, they are generated and manipulated with the following programming code (See Figure 1) to execute Green's function

Figure 1. Definition of the structure of the data matrix and definition of the Green function.

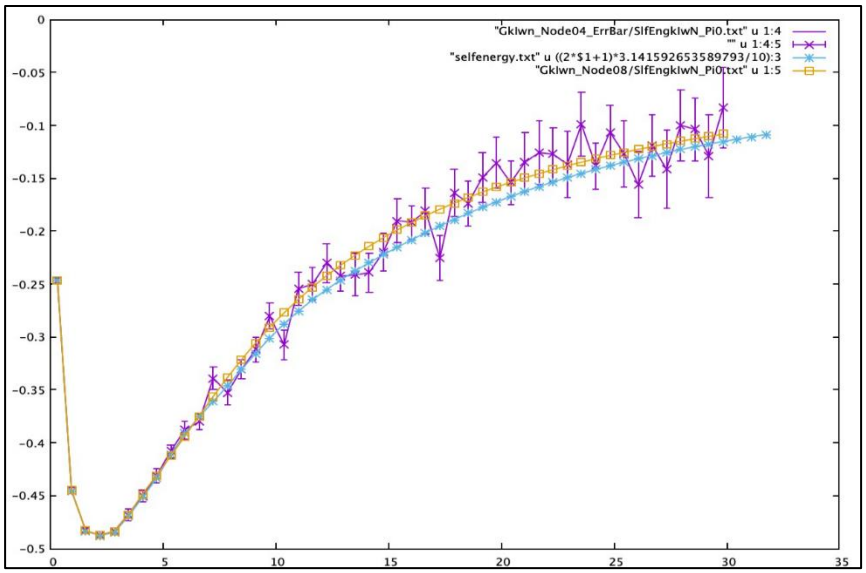
```
using GreenFunc

β = 100.0; E = 1.0 # inverse temperature and the level energy
ωₙ_mesh = MeshGrids.ImFreq(100.0, FERMION; Euv = 100E) # UV energy cutoff is 100 times larger than the level energy
Gn =  MeshArray(ωₙ_mesh; dtype=ComplexF64); # Green's function defined on the ωₙ_mesh

for (n, ωₙ) in enumerate(Gn.mesh[1])
    Gn[n] = 1/(ωₙ*im - E)
end
```

Green's function numerically describes the correlations between two or more spacetime events. The space-time continuum needs to be discretized into spatial and temporal meshes. This example demonstrates how to define a Green function of a body in a temporal mesh. The package provides three types of temporal meshes: imaginary time grid, Matsubara frequency grid, and (Chikano et al., 2018; By Tullio, 2021) Lehmann discrete representation grid (DLR). The latter provides a generic compressed representation for Green functions (how to use DLR will be shown later). As a result, the ImTime, ImFreq, and DLRFreq methods can be created. So it is required to specify the reverse temperature, whether the particle is fermion or boson (defined using the Fermion or Boson constant). Internally, a set of non-uniform grid points optimized for the given reverse temperature and cut-off energy with the given parameters will be created.

Once the meshes are created, you can define a MeshArray function on top of them to represent the Green Gn function. It should be noted that one can through MeshArray taker a set of meshes and initialize a multidimensional matrix, where eachmesh corresponds to a dimension of the matrix. The MeshArray data type is specified by the argument using the optional keyword dtype, which is set to Float64 by default. You can access meshes (stored as a tuple) with Gn.mesh and array data with Gn.data.

By default, Gn.data is left undefined if you do not specify it in the programming line. To initialize it, you can use the optional keyword argument data or use the iterating interface of the meshes and the MeshArray.

Figure 2. Numerical comparison of the energy potential as a function of the meshes for the Green function



4.2. Illustration of Green's function for free electron gas

In this case study, we show how to create a Green function of a gas of free electrons. Unlike the spinless fermionic particle, the electron is a particle of spin 1/2, so it has two internal states. In free space it has a kinetic energy (By Tullio, et al., 2019):

$$\epsilon_q = q^2 - E \qquad (5)$$

(Drive is used $m_e = {}^1/_2$)

Green's function, in the frequency space of Matsubara, is given by the following equation:

$$G_n = G_{\sigma_1,\sigma_2}(q, i\omega_n) = \frac{1}{i\omega_n - \epsilon_q} \qquad (6)$$

Donde denotes the spins of the incoming and outgoing electron in the propagator. σ_iThe Matsubara frequency grid in the first illustration is shown (See Figure 1). It shows how the CompositeGrids package should be used to generate the momentum grids and how to handle the multiple internal states andmeshes generated by the MeshArray function.

Figure 3. Green's function of a discrete internal state free electron gas 2x2

```
using GreenFunc, CompositeGrids
β = 100.0; E = 1.0 # inverse temperature and the level energy
ωₙ_mesh = MeshGrids.ImFreq(100.0, FERMION; Euv = 100E) # UV energy cutoff is 100 times larger than the level energy
kmesh = SimpleGrid.Uniform{Float64}([0.0, 10.0], 50); # initialze an uniform momentum grid
G_n =  MeshArray(1:2, 1:2, kmesh, ωₙ_mesh; dtype=ComplexF64); # Green's function of free electron gas with 2x2 innerstates

for ind in eachindex(G_n)
    q = G_n.mesh[3][ind[3]]
    ω_n = G_n.mesh[4][ind[4]]
    G_n[ind] = 1/(ω_n*im - (q^2-E))
end
```

Various types of grids can be generated with the CompositeGrids package. This SimpleGrid module provides several basic grids, such as uniform grids and logarithmically dense grids. The method implementation (Uniform) generates a linearly spaced 1D grid. It is necessary to specify the number of points in the N grid and the contour points [min, max]. Also, arbitrary numbers can be combined with the SimpleGrid module for sub-grids with a user-specified pattern definedby a grid panel file.

By using MeshArray, it can take any iterable object as one of its meshes. Therefore, for discrete internal states such as spins, one can simply use 1:2, which is a UnitRange{Int64}object.

4.3. Illustration of the Green function for a Hubbard network

This illustration begins, on how to generate a multidimensional Green function in a Brillouin Zone mesh. (Rubtsov et al., 2008). The Green's function of a spinless free Fermi gas in a square lattice is calculated.

It has atight junction dispersion (Essler et al., 2005):

$$\epsilon_q = -2t\big(cos(q_x) + \cos\big(q_y\big)\big) \quad (7)$$

Lwhat gives:

$$G_n(q, \omega_n) = \frac{1}{i\omega_n - \epsilon_q} \quad (8)$$

The impulse is defined in the first Brillouin zone captured by a 2D k-mesh.

The dimension of these matrices defined as 2D, becomes manageable by the speed of Julia to simulate systems with fixed number of particles in the corresponding sube space, which must allow to reach dimensions much higher than defining an entire space. (Kaye et al., 2022)

Figure 4. Multidimensional Green function in a Brillouin Zone mesh

```
using GreenFunc
using GreenFunc: BrillouinZoneMeshes

DIM, nk = 2, 8
latvec = [1.0 0.0; 0.0 1.0] .* 2π
bzmesh = BrillouinZoneMeshes.BaseMesh.UniformMesh{DIM, nk}([0.0, 0.0], latvec)
ωₙmesh = ImFreq(10.0, FERMION)
g_freq =  MeshArray(bzmesh, ωₙmesh; dtype=ComplexF64)

t = 1.0
for ind in eachindex(g_freq)
    q = g_freq.mesh[1][ind[1]]
    ωₙ = g_freq.mesh[2][ind[2]]
    g_freq[ind] = 1/(ωₙ*im - (-2*t*sum(cos.(q))))
end
```

For lattice systems with multidimensional Brillouin zone, momentum grids are generated internally with the BrillouinZoneMeshes.jl package. Here a UniformMesh{DIM,N}(origin, latvec) generates a linearly spaced momentum mesh in the first Brillouin zone defined by the origin and the given lattice vectors.

4.4. Illustration on the Fourier transform of the Green function with discrete Lehmann representation (DLR)

The compatible form based on the discrete representation of Lehmann (DLR) is developed in the Lehmann.jl package. , which provides a compact representation for the Green functions of a field . (Kaye et al., 2022)

At this point, we start from the numerical definition for temperature and a T level of precision, ϵassuming a generic Green function with only: basic functions labeled by a set of grid points of real frequency. $\log(1/T) \log (1/\epsilon)$

The above, to execute the fast Fourier transform and interpolation between imaginary time with Matsubara frequency representations (Using the Composite Grids.jl package.) with a cost$O\left(log(1/T)log(1/\epsilon)\right)$.

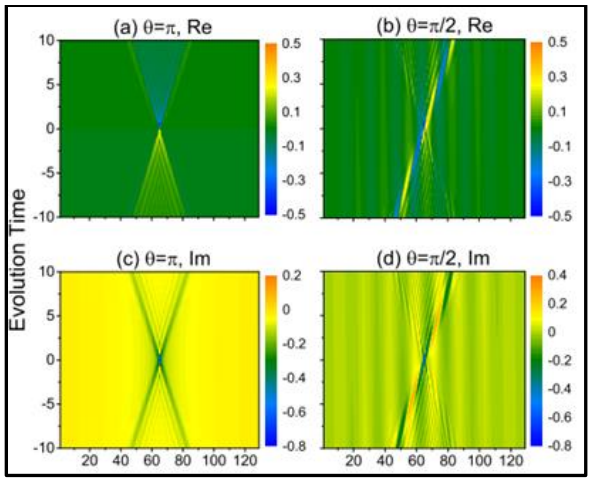
In these terms, we proceed to illustrate in figure 4, how the Fourier transform based on DLR GreenFunc.jl can be realized between imaginary time and Matsubara frequency domains from one side to the other through the DLR representation. (Kaye et al., 2022)

Figure 5. DLR-based Fourier transform between imaginary time and Matsubara frequency domains

```
using GreenFunc, CompositeGrids

β = 100.0; E = 1.0 # inverse temperature and the level energy
ωₙ_mesh = ImFreq(100.0, FERMION; Euv = 100E) # UV energy cutoff is 100 times larger than the level energy
kmesh = SimpleGrid.Uniform{Float64}([0.0, 10.0], 50); # initialze an uniform momentum grid
G_n =  MeshArray(1:2, 1:2, kmesh, ωₙ_mesh; dtype=ComplexF64); # Green's function of free electron gas with 2x2 innerstates

for ind in eachindex(G_n)
    q = G_n.mesh[3][ind[3]]
    ω_n = G_n.mesh[4][ind[4]]
    G_n[ind] = 1/(im*ω_n - (q^2-E))
end

G_dlr = to_dlr(G_n) # convert G_n to DLR space
G_tau = to_imtime(G_dlr) # convert G_dlr to the imaginary-time domain
```

According to the command routine defined in Figure 5, we have proceeded to specify the parameters of the system: Tinverse emperature and the energy level, the length of the chain (See Figure 5)

Figure 6: The real (top panel) and imaginary (bottom panel) part of the larger Green function for θ = π (left panel) and θ = π/2 (right panel).



The Green function at imaginary time, after the Fourier transform, must be consistent with the following analytic solution:

$$G_r = -\frac{e^{-r\epsilon_q}}{\left(1+e^{-\beta\epsilon_q}\right)} \qquad (9)$$

For any Green function that has at least one imaginary-temporal mesh cell (ImTime, ImFreq, and DLRFreq), a set of operations (to_dlr, to_imfreqy to_imtime) is provided to connect the DLR space with theimaginary time and the Matsubara frequency space. By default, all these functions find in the dimension of the imaginary-temporal mesh within the meshes of Green functions and perform the transformation with respect to it. Alternatively, you can specify the dimension with the optional keyword argument dim. Even, care must be taken thatthe original version of DLRonly works with the Green's function of a single body.

Once a G_dlr spectral density is obtained in DLR space, functions (to_imfreq or to_imtime) that employ methods can be used to reconstruct the Green's function in the corresponding space. By default, to_imfreq and to_imtime uses an optimized grid of imaginary time or Matsubara frequency of the DLR. (Kaye et al., 2022)An imaginary time target or Matsubara frequency grid can be assigned if necessary.

Combining to_dlr, to_imfreqy, to_imtime must allowto go, both interpolation and Fourier transform. Since the spectral density G_dlr, it can be reused whenever you tend to change the grid points of the Green function (usually through interpolation that loses more precision than the DLR transformation), it is recommended that you always keep the objects G_dlr. In the case of not needingr the DLR Green intermediate function, the (Shinaoka et al. 2017.)operator can be used as |>shownin Figure 4 to make the Fourier transform directlybetween ImFreq and ImTime.

5. Discussion

The development of open source libraries has promoted, in recent years, many ideas and have establishedsolid foundations for future lines of research where the use of these computational tools prevails that emerges in a more democratic and efficient science. However, there are still many unexplored areas that would benefit fromsuch developments. Within these, quantum information with free electrons, fermions, Hubbard lattice and application of Fourier transform can be included, as developed in this article.

Additionally, the implementation of the GreenFunc.jl library under the Julia language is relevant since it represents a high-level language with very high speed, which is growing remarkably in the field of scientific computing. In addition, Julia has been chosen because it has a performance 100 times higher than Python in the same task, being also equal (Xiu-Zhe et al., 2020)or easier to read and write.

6. Conclusions

The magnetic properties of matter are essentially a consequence of electrons (free or bound), of their magnetic moments, where the study of the magnetic properties of an (ideal) gas of electrons is relevant in this context. Therefore, this article based on representing the operation of the open source library GreenFunc.jl developed by different collaborators under the programming language Julia, was very useful as tools of numerical order to own to simulate free electron systems, Fermionics and spinless free Fermi gas systems. In addition, these tools are useful not only in the area of quantum information, but also in condensed matter, nanomaterials and quantum chemistry. Although, it should be noted that there are other fermionic libraries, such as OpenFermion; however, GreenFunc.jl was taken into consideration because it has a primarily orbital-based approach. In this sense, (Jarrod et al., 2020)Fermionic.jl defines fermionic operators for finite dimension as matrices, and states as vectors on which to operate, which is much more suitable for quantum information protocols. However, the greatest advantage of the library is the ability to work in systems(By Tullio, 2021)with fixed particle numbers.

Bibliography

Ar. Abanov et al. . (17 de Junio de 2022). Quantum-critical theory of the spin-fermion model and its application to cuprates. Normal state analysis. arXiv, 1-223. Obtenido de https://www.arxiv-anity.com/papers/cond-mat/0107421/

Beenakker et al. (2004). Charge detection enables free-electron quantum computation. . Phys. Rev. Lett. 020501 , 93. doi:https://doi.org/10.1103/PhysRevLett.93.020501

Besard et al. (2017). Effective extensible programming: Unleashing julia on gpus. . CoRR, abs/1712.03112. doi:10.1109/tpds.2018.2872064

Chikano et al. (2018). Performance analysis of a physically constructed orthogonal representation of imaginary-time Green's function. Phys. Rev. B, 98(3), 035104. Obtenido de http://s-read.saitama-u.ac.jp/researchers/pages/researcher_en/YQWpVZvb

Chikano et al. (2019). irbasis: Open-source database and software for intermediaterepresentation basis functions of imaginary-time Green's function. Comput. Phys. Commun., 240, 181–188. doi:https://doi.org/10.1016/j.cpc.2019.02.006

Di Tullio. (2021). Entanglement and correlations in fermionic systems. Doctoral Thesis Work. National University of La Plata. doi :https://doi.org/10.35537/10915/130069

Di Tullio. (2021). Toolkit for fermonic simulations and fermionic quantum computation in Julia. Obtenido de https://github.com/Marco-Di-Tullio/Fermionic.jl

Di Tullio, et al. . (03 de Diciembre de 2019). Fermionic entanglement in the Lipkin model. . Phys. Rev. A , 062104. doi:10.1103/PhysRevA.100.062104

Essler et al. . (2005). The one-dimensional Hubbard model. Cambridge University Press. Obtenido de http://insti.physics.sunysb.edu/~korepin/PDF_files/Hubbard.pdf

Gigena et al. . (2020). One-body entanglement as a quan- tum resource in fermionic systems. . Phys. Rev A 042410., 102. doi:https://doi.org/10.1103/PhysRevA.102.042410

Gigena, et al. (2021). Many-body entanglement in fermion systems. . Phys. Rev. 052424, 103. doi:https://doi.org/10.1103/PhysRevA.103.052424

Jarrod et al. (2020). OpenFermion: the electronic structure package for quantum computers. Quantum Science and Technology , 5(034014). Obtenido de https://iopscience.iop.org/article/10.1088/2058-9565/ab8ebc

Kaye et al. (18 de Febrero de 2022). Discrete Lehmann representation of imaginary time Green's functions. arXiv:2107.13094. doi:https://doi.org/10.48550/arXiv.2107.13094

Kaye et al. (2022). Libdlr: Efficient imaginary time calculations using the discrete Lehmann representation. arxiv. Obtenido de https://arxiv.org/pdf/2110.06765.pdf

Rubtsov et al. (2008). Dual fermion approach to nonlocal correlations in the Hubbard model. Phys. Rev. 033101, 77. doi:https://doi.org/10.1103/physrevb.77.033101

Schüler and Pavlyukh. (2018). Spectral properties from Matsubara Green’s function approach: Application to molecules. Phys. Rev. B , 97, 115164. Obtenido de https://aip.scitation.org/doi/abs/10.1063/5.0003145

Shinaoka et al. 2017. (s.f.). Compressing Green’s function using intermediate representation between imaginary-time and real-frequency domains. . Phys. Rev. B,., 96(3). doi:https://doi.org/10.1103/PhysRevB.96.035147

Tosta et al. (2019). Quantum computation from fermionic anyons on a one-dimensional lattice. Phys. Rev, A 062335, 99. doi:https://doi.org/10.1103/PhysRevA.99.062335

Wenhu et al. (17 de March de 2017). Quantum Critical Point revisited by the Dynamical Mean Field Theory . Phys. Rev. 121113(R) , 95. doi:https://doi.org/10.1103/PhysRevB.95.121113

Xiu-Zhe et al. . (2020). Extensible, Efficient Framework for Quantum Algorithm Design. Quantum, 4, 341. doi:https://doi.org/10.22331/q-2020-10-11-341