Time-frequency component of the GreenX library: minimax grids for efficient RPA and GW calculations

Maryam Azizi 1, Jan Wilhelm 2, Dorothea Golze 3, Matteo Giantomassi 1, Ramón L. Panadés-Barrueta 3, Francisco A. Delesma 4, Alexander Buccheri 5, Andris Gulans 6, Patrick Rinke 4, Claudia Draxl 5, and Xavier Gonze 1

1 Institute of Condensed Matter and Nanoscience, UCLouvain, B-1348 Louvain-la-Neuve, Belgium
2 Institute of Theoretical Physics and Regensburg Center for Ultrafast Nanoscopy (RUN), University of Regensburg, D-93053 Regensburg, Germany
3 Faculty of Chemistry and Food Chemistry, Technische Universität Dresden, 01062 Dresden, Germany
4 Department of Applied Physics, Aalto University, P.O. Box 11100, 00076 Aalto, Finland
5 Institut für Physik und Iris Adlershof, Humboldt-Universität zu Berlin, Zum Großen Windkanal 2, 12489 Berlin, Germany
6 Department of Physics, University of Latvia, Jelgavas iela 3, Riga, LV-1004 Latvia

DOI: 10.21105/joss.05570

Software

- Review
- Repository
- Archive

Editor: Lucy Whalley

Reviewers:
- @mailhexu
- @DarioALeonValido

Submitted: 24 May 2023
Published: 03 October 2023

License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License (CC BY 4.0).

Summary

The central objects in many-body electronic structure theory, such as the GW method and the random-phase approximation (RPA), are defined in the complex frequency or time domain. We present here the GX-TimeFrequency component of the GreenX library, providing grids and weights for imaginary time-frequency transformations needed for Green’s function based objects. The GreenX library emerged from the NOMAD Center of Excellence, whose objective is to enable accurate Green’s function based electronic structure theory calculations on state-of-the-art supercomputers.

The package comprises minimax time and frequency grids (Kaltak et al., 2014b; Liu et al., 2016; Takatsuka et al., 2008) and corresponding quadrature weights to numerically compute time and frequency integrals of the correlation energy as well as weights for Fourier transforms between time and frequency grids. While targeting low-scaling RPA and GW algorithms, its compact frequency grids allows one to reduce the computational prefactor in RPA implementations with conventional scaling. In addition, the time grids can be employed in Laplace-transformed direct MP2 (LT-dMP2) calculations. The GreenX source code is freely available on GitHub, and comes equipped with a build system, a comprehensive set of tests, and detailed documentation.

Statement of need

RPA, an accurate approach to compute the electronic correlation energy, is non-local, includes long-range dispersion interactions and dynamic electronic screening, and is applicable to a wide range of systems from 0 to 3 dimensions (Eshuis et al., 2012; Ren, Rinke, Joas, et al., 2012). The GW method (Hedin, 1965) is based on the RPA susceptibility and has become the method of choice for the calculation of direct and inverse photoemission spectra of molecules and solids (Golze et al., 2019; Reining, 2018). Furthermore, GW forms the basis for Bethe-Salpeter Equation (BSE) calculations of optical spectra (Onida et al., 2002).

Despite wide adoption, RPA and GW face computational challenges, especially for large systems. Conventional RPA and GW implementations scale with the fourth power of system size $N$ and are therefore usually limited to systems up to one hundred atoms (Panadés-Barrueta & Golze, 2023; Stuke et al., 2020). To tackle larger and more realistic systems, scaling
reductions present a promising strategy to decrease the computational cost. Such low-scaling algorithms utilize real-space representations and time-frequency transformations, such as the real-space/imaginary-time approach (Rojas et al., 1995) that reduces the complexity to $O(N^3)$. Several such cubic-scaling $GW$ algorithms have recently been implemented, e.g. in a plane-wave/projector-augmented-wave $GW$ code (Kutepov et al., 2017; Liu et al., 2016) or with localized basis sets using Gaussian (Duchemin & Blase, 2021; Grami et al., 2023; Wilhelm et al., 2018, 2021) or Slater-type orbitals (Förster et al., 2023; Förster & Visscher, 2020, 2021a, 2021b). Similarly, low-scaling RPA algorithms were implemented with different basis sets (Drontschenko et al., 2022; Duchemin & Blase, 2019; Graf et al., 2018; Kaltak et al., 2014a, 2014b; Shi et al., 2023; Wilhelm et al., 2016).

With larger pre-factor, low-scaling algorithms are typically more expensive for smaller systems and only become more cost-effective than canonical implementations for larger systems thanks to their reduced scaling (Wilhelm et al., 2018). Furthermore, the numerical precision of low-scaling $GW$ algorithms is strongly coupled to the time-frequency treatment (Wilhelm et al., 2021). Early low-scaling $GW$ algorithms did not reach the same precision as canonical implementations (Förster & Visscher, 2020; Vlcek et al., 2017; Wilhelm et al., 2018). Although appropriate Fourier transforms and corresponding time-frequency grids have been implemented (Duchemin & Blase, 2021; Förster & Visscher, 2021a; Liu et al., 2016; Wilhelm et al., 2021), these implementations and grids are tied to particular codes and are often buried deeply inside the code. Furthermore, reuse of such implementations elsewhere is often restricted by license requirements or dependencies on definitions made in the host code.

In this work, we present the GX-TimeFrequency component of the GreenX library, an open-source package distributed under the Apache license (Version 2.0). GX-TimeFrequency provides time and frequency grids and corresponding integration weights to compute correlation energies for Green’s function implementations. It also provides Fourier weights to convert between imaginary time and imaginary frequency. The library can be used for low-scaling RPA and $GW$ implementations, or BSE codes, which use (low-scaling) $GW$ as input. The minimax grids are also suitable for RPA implementations with conventional scaling (Del Ben et al., 2015): they are more compact than, e.g., Gauss-Legendre grids, resulting in a reduction of the computational prefactor, while yielding same accuracy (Del Ben et al., 2015). However, minimax grids are not recommended for conventional imaginary-frequency-only $GW$ implementations (Ren, Rinke, Blum, et al., 2012) since they have not been optimized for the frequency integral of the self-energy.

While not being the main target of the library, the minimax time grids can also be utilized to calculate the LT-dMP2 correlation energy (Almlöf, 1991; Glasbrenner et al., 2020; Jung et al., 2004; Kaltak et al., 2014b; Takatsuka et al., 2008). The dMP2 term is one of two terms of the MP2 correlation energy and, in a diagrammatic representation, corresponds to the lowest order of the RPA correlation energy (Ren, Rinke, Joas, et al., 2012). The dMP2 correlation energy can be reformulated using the Laplace transform to obtain the LT-dMP2 expression which scales cubically in contrast to the $O(N^5)$ scaling of standard MP2.

Mathematical framework

The single-particle Green’s function $G$ and the non-interacting susceptibility $\chi^0$ are starting points for several many-body perturbation theory methods. In canonical implementations, $\chi^0(r, r', i\omega)$ is often expressed in the Adler-Wiser form (Adler, 1962; Wiser, 1963), where the sums over occupied (index $j$) and unoccupied (index $a$) single-particle states $\psi$ are coupled via their corresponding energies $\varepsilon$.
Figure 1: Sketch of the methods supported by GX-TimeFrequency which start from $\tilde{\chi}_0(i\tau)$. In addition to the discrete time and frequency grids $\{\tau_j\}$ and $\{\omega_k\}$, the library provides the corresponding weights $\{\sigma_j\}$ and $\{\gamma_k\}$ for the integration of the correlation energy $E_c$ as well as the Fourier weights $\delta_{kj}$, $\eta_{jk}$ and $\lambda_{jk}$ defined in Equations 2–4. The bare and screened Coulomb interactions are indicated by $v(r, r') = 1/|r - r'|$ and $W(i\omega)$, respectively. $\epsilon(i\omega)$ is the dynamical dielectric function, $\Sigma$ the GW self-energy, and AC stands for analytic continuation.

The Adler-Wiser expression of $\chi_0^0(i\omega)$ can be transformed into the imaginary time domain, $\tilde{\chi}_0^0(r, r', i\tau) = -iG(r, r', i\tau)G(r', r, -i\tau)$, yielding the equation in the yellow box in Fig. 1, where the two sums are separated, leading to a favorable $O(N^3)$ scaling. The polarizability $\tilde{\chi}_0^0(i\tau)$ is the starting point for LT-dMP2 and low-scaling RPA and GW. The low-scaling GW procedure shown in Fig. 1 is known as the space-time method and given here in its original formulation for planewave codes (Rojas et al., 1995).

The time-frequency integrals in Fig. 1 are performed numerically. All three methods in Fig. 1 require a discrete time grid $\{\tau_j\}_{j=1}^n$, where $n$ is the number of grid points. RPA and GW additionally need the discrete frequency grid $\{\omega_k\}_{k=1}^n$. Since $\tilde{\chi}_0^0(r, r', i\tau)$ is sharply peaked around the origin and then decays slowly, homogeneous time and frequency grids are inefficient. For this reason, non-uniform grids like Gauss-Legendre (Rieger et al., 1999), modified Gauss-Legendre (Ren, Rinke, Blum, et al., 2012) and the here presented minimax (Kaltak et al., 2014b) grids are used. The minimax grids include also integration weights for the computation of the correlation energies. For the calculation of the LT-dMP2 correlation energy $E_c^{\text{dMP2}}$ (Kaltak et al., 2014b; Takatsuka et al., 2008), a time quadrature is performed, for which our library provides the integration weights $\{\sigma_j\}_{j=1}^n$. Similarly, the RPA correlation energy $E_c^{\text{RPA}}$ (Del Ben et al., 2015; Kaltak et al., 2014b) is computed from frequency quadrature using integration weights $\{\gamma_k\}_{k=1}^n$.

Low-scaling RPA and GW algorithms include the Fourier transform of $\tilde{\chi}_0^0(i\tau)$ to $\chi_0^0(i\omega)$ (blue dashed box in Fig. 1). The GW space-time method performs two additional Fourier transforms: The screened Coulomb interaction $W(i\omega)$ is transformed to imaginary time (red dashed box), and the self-energy $\tilde{\Sigma}(i\tau)$ is Fourier transformed back to imaginary frequency (green dashed box).

The conversion between imaginary time and frequency grids relies on nonuniform discrete cosine and sine transformations for even and odd functions $F^{\text{even/odd}}$, respectively (Liu et al., 2016). If the function $F$ is neither odd nor even, the computation of functions $\tilde{F}(i\tau)$ and $F(i\omega)$ is split into even and odd parts (Liu et al., 2016)

$$\tilde{F}(i\tau) = F^{\text{even}}(i\tau) + F^{\text{odd}}(i\tau) \quad \text{and} \quad F(i\omega) = F^{\text{even}}(i\omega) + F^{\text{odd}}(i\omega)$$

with $F^{\text{even}}(x) = F^{\text{even}}(-x)$ and $F^{\text{odd}}(x) = -F^{\text{odd}}(-x)$. The same parity rules hold for quantities with a hat. The corresponding discrete Fourier transforms read (Liu et al., 2016)

The optimal grid parameters \( \tau_j, \sigma_j, \omega_k, \gamma_k, \delta_{kj}, \eta_{jk}, \lambda_{kj} \) depend on the energy gap \( \Delta_{\text{min}} := \min(\varepsilon_a - \varepsilon_j) \) and the maximum eigenvalue difference \( \max(\varepsilon_a - \varepsilon_j) \) of the material. We generated minimax grid parameters \( \tau_j, \sigma_j, \omega_k, \gamma_k \) assuming energy differences \( \varepsilon_a - \varepsilon_j \in [1, R] \), see details in Refs. (Hackbusch, 2019; Kaltak et al., 2014b). Our library stores minimax grid parameters \( \{\tau_j(R)\}_{j=1}^{n}, \{\sigma_j(R)\}_{j=1}^{n}, \{\omega_k(R)\}_{k=1}^{n}, \{\gamma_k(R)\}_{k=1}^{n} \) for \( n \in [6, 34] \) and for different values of the range \( R \) (on average 15 \( R \)-values for each \( n \)). For a material with energy gap \( \Delta_{\text{min}} := \min(\varepsilon_a - \varepsilon_j) \) and maximum eigenvalue difference \( \Delta_{\text{max}} := \max(\varepsilon_a - \varepsilon_j) \), one easily obtains the material-targeted minimax parameters \( \{\tau_j^{\text{mat}}\}_{j=1}^{n}, \{\sigma_j^{\text{mat}}\}_{j=1}^{n}, \{\omega_k^{\text{mat}}\}_{k=1}^{n}, \{\gamma_k^{\text{mat}}\}_{k=1}^{n} \) from rescaling stored parameters with a range \( R \geq \Delta_{\text{max}} / \Delta_{\text{min}} \) (Hackbusch, 2019; Kaltak et al., 2014b),

\[
\omega_k^{\text{mat}} = \Delta_{\text{min}} \omega_k(R), \quad \gamma_k^{\text{mat}} = \Delta_{\text{min}} \gamma_k(R), \quad \tau_j^{\text{mat}} = \frac{\tau_j(R)}{2\Delta_{\text{min}}}, \quad \sigma_j^{\text{mat}} = \frac{\sigma_j(R)}{2\Delta_{\text{min}}}
\]

Required input and output

GX-TimeFrequency requires as input the grid size \( n \), the minimal eigenvalue difference \( \Delta_{\text{min}} \), and the maximal eigenvalue difference \( \Delta_{\text{max}} \). For the output parameters, see Table 1. The library component retrieves tabulated minimax parameters \( \{\tau_j(R)\}_{j=1}^{n}, \{\sigma_j(R)\}_{j=1}^{n}, \{\omega_k(R)\}_{k=1}^{n}, \{\gamma_k(R)\}_{k=1}^{n} \) of the requested grid size \( n \) for the smallest range \( R \) that satisfies \( R \geq \Delta_{\text{max}} / \Delta_{\text{min}} \). GX-TimeFrequency then rescales the retrieved minimax parameters according to Equation 6 with \( \Delta_{\text{min}} \) and prints the results \( \{\tau_j^{\text{mat}}\}_{j=1}^{n}, \{\sigma_j^{\text{mat}}\}_{j=1}^{n}, \{\omega_k^{\text{mat}}\}_{k=1}^{n}, \{\gamma_k^{\text{mat}}\}_{k=1}^{n} \). Fourier integration weights are computed on-the-fly via least-squares optimization. The precision of a global forward cosine transformation followed by backward cosine transformations, is measured

\[
F^{\text{even}}(\omega_k) = \sum_{j=1}^{n} \delta_{kj} \cos(\omega_k \tau_j) F^{\text{even}}(i \tau_j) \quad F^{\text{odd}}(i \tau_j) = i \sum_{j=1}^{n} \lambda_{kj} \sin(\omega_k \tau_j) F^{\text{odd}}(i \tau_j)
\]

\[
F^{\text{even}}(i \tau_j) = \sum_{k=1}^{n} \eta_{jk} \cos(\tau_j \omega_k) F^{\text{even}}(i \omega_k) 
\]

\[
F^{\text{odd}}(i \tau_j) = -i \sum_{k=1}^{n} \xi_{jk} \sin(\tau_j \omega_k) F^{\text{odd}}(i \omega_k)
\]
from

\[ \Delta_{CT} = \max_{j,j' \in \{1,2,\ldots,n\}} \left| \sum_{k=1}^{n} \eta_{j'k} \cos(\tau_{j'}\omega_{k}) \cdot \delta_{kj} \cos(\omega_{k}\tau_{j}) - (I)_{j'j} \right| \]  

with \( I \) being the identity matrix. Inputs and outputs are in atomic units.

Table 1: Output returned by the GX-TimeFrequency component of GreenX. We abbreviate low-scaling as ls, and least-squares optimization as L2 opt.

<table>
<thead>
<tr>
<th>Output</th>
<th>Description</th>
<th>Methods using the output</th>
<th>Computation</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \tau_{j} ), ( \nu_{j} ), ( \omega_{k} ), ( \delta_{kj} ), ( \eta_{jk} ), ( \lambda_{kj} )</td>
<td>time points, integration weights, frequency points, Fourier weights, Fourier weights, Fourier weights</td>
<td>LT-dMP2, ls, RPA, ls, GW tabulated + rescaling</td>
<td>LT-dMP2, ls, canonical RPA, ls, GW tabulated + rescaling</td>
</tr>
<tr>
<td>( \Delta_{CT} )</td>
<td>duality error</td>
<td>ls, GW</td>
<td>on-the-fly</td>
</tr>
</tbody>
</table>

Acknowledgements

Work supported by the European Union’s Horizon 2020 research and innovation program under the grant agreement N° 951786 (NOMAD CoE). J.W. and D.G. acknowledge funding by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) via the Emmy Noether Programme (Project No. 503985532 and 453275048, respectively). A.G. acknowledges funding provided by the European Regional Development Fund via the Central Finance and Contracting Agency of Republic of Latvia (Project No. 1.1.1.5/21/A/004).

References


