AutoBZ.jl

Adaptive Brillouin zone integration for optical conductivity

Lorenzo X. Van Muñoz[1]

[1]Department of Physics
Massachusetts Institute of Technology

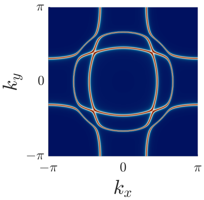
Feb. 2, 2024
subMIT Workshop

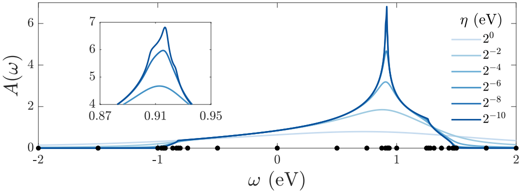
Brillouin zone integration

Brillouin zone integrals are costly and have localized integrands.
For example, the single-particle retarded Green's function,

$$G(\omega) = \int_{\mathrm{BZ}} dk \ \mathrm{Tr}\left[\left(\omega + \mu - H(k) - \Sigma(\omega)\right)^{-1}\right].$$

In applications such as DMFT, $\mathrm{Im}(\Sigma(\omega)) > 0$ mollifies the integrand.
WLOG $\Sigma(\omega) = -i\eta$, where η sets the scale of localized features.





Kaye et al. (2023)

Manuscript in review

Brillouin zone integration methods

Standard methods

- Periodic Trapezoidal Rule (PTR)
  note: *spectrally* convergent
- Tetrahedron methods
- Adaptive smearing methods

h/p–Adaptive methods

- Gauss-Legendre quadrature
  note: high-order convergence

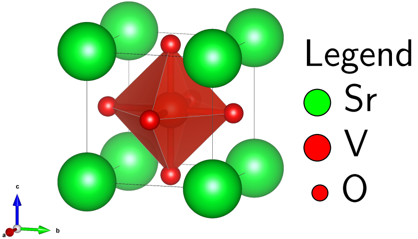
Point: High-order quadrature rules get more digits of accuracy from integrand evaluations

Observation: Wannier interpolation enables fast, on-the-fly integrand evaluations

Optical conductivity of $SrVO_3$ Fermi liquid

Strontium Vanadate is a cubic transition metal perovskite commonly used for algorithmic benchmarks.

We use Fermi liquid scaling for $\eta \sim T^2$

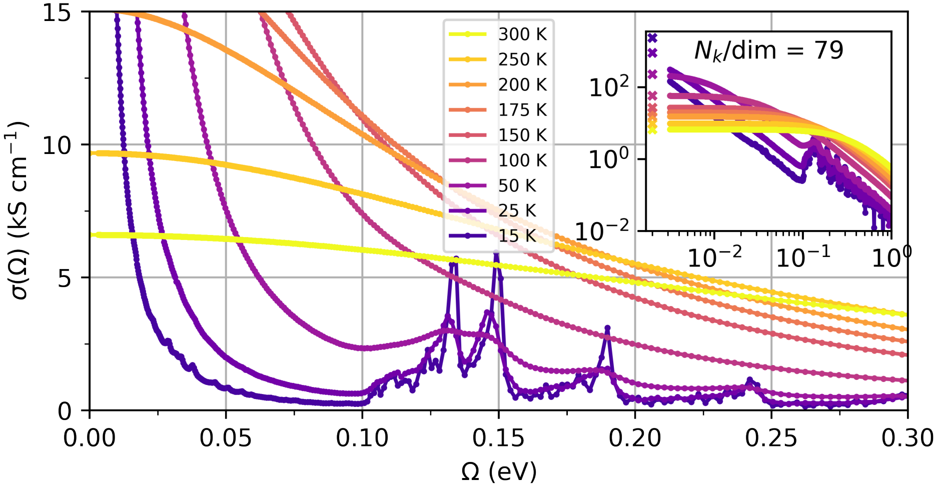


$$\Gamma_{\alpha\beta}(\omega_1, \omega_2) = \int_{\mathrm{BZ}} d\mathbf{k} \ \mathrm{Tr}[v_\alpha(\mathbf{k}) A(\mathbf{k}, \omega_1) v_\beta(\mathbf{k}) A(\mathbf{k}, \omega_2)]$$

$$\sigma_{\alpha\beta}(\Omega) = \pi e^2 \hbar \int_{-\infty}^{\infty} d\omega \ \frac{f(\omega) - f(\omega + \Omega)}{\Omega} \Gamma_{\alpha\beta}(\omega, \omega + \Omega)$$

https://triqs.github.io/dft_tools/latest/guide/transport.html

Problem: convergence for localized integrands



Manuscript in preparation (2024)

Solution: automatic and adaptive algorithms

We compare adaptive algorithms with

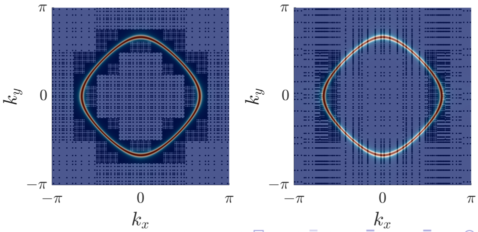
- automatic convergence to user-specified error tolerance
- high-order convergence

Algorithms

- PTR: p-adaptive
  note: easiest for large η
- TAI: h-adaptive
- IAI: h-adaptive
  note: best $\eta \to 0^+$ scaling

Method	Complexity
PTR	$\mathcal{O}(\eta^{-d})$
TAI	$\mathcal{O}(\log(\eta^{-1})/\eta^{d-1})$
IAI	$\mathcal{O}(\log^d(\eta^{-1}))$

Kaye et al. (2023) Manuscript in review

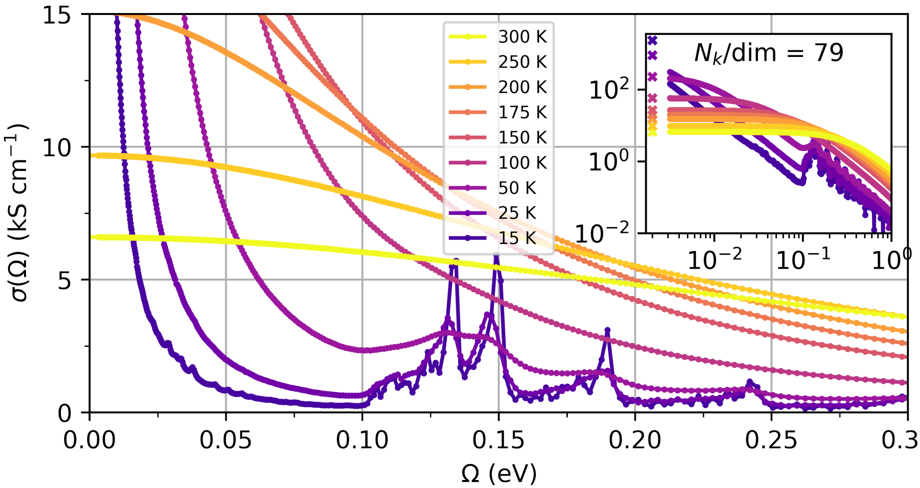


Introducing AutoBZCore.jl

- Pure-Julia package dedicated to (BZ) integrals
- IAI+PTR optimized for Wannier(90) interpolation
- User-defined integrals derived from Integrals.jl
- IBZ integration using SymmetryReduceBZ.jl
- Parallelized batch interfaces, with HDF5 output
- Works from Python, MATLAB

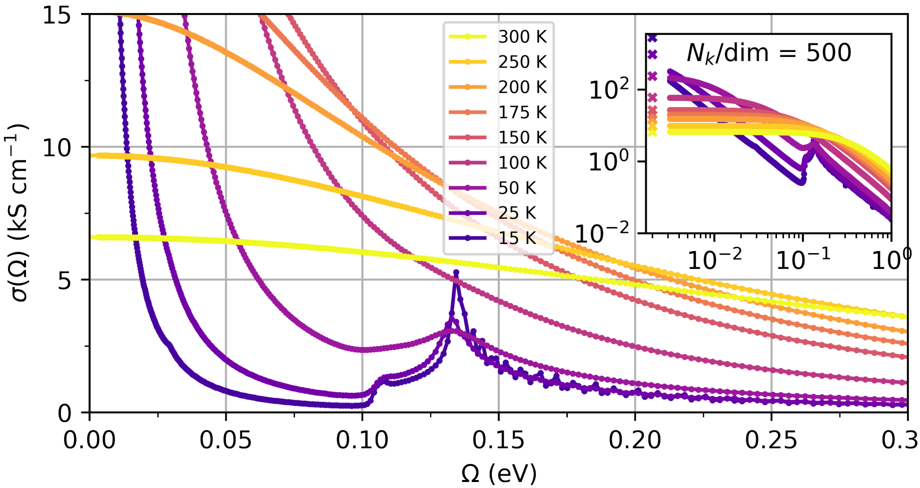
Similar functionality to `wannierberri`, better algorithms

Optical conductivity of $SrVO_3$ Fermi liquid



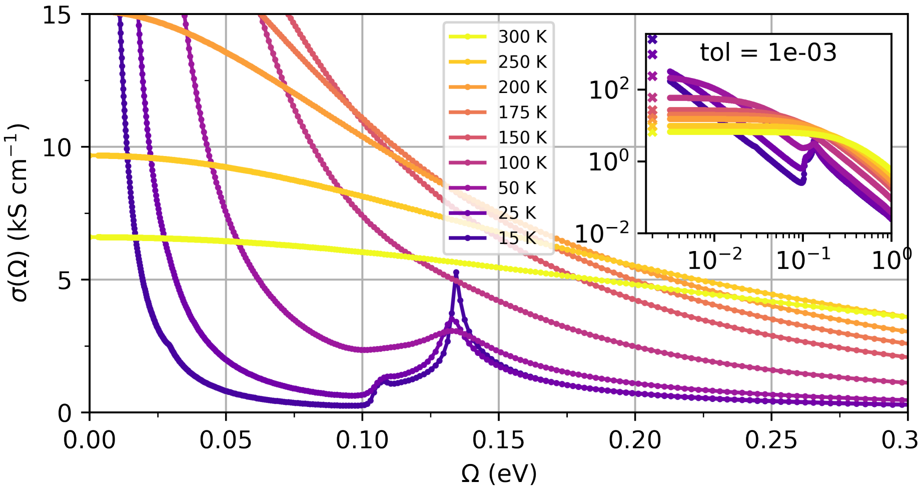
Manuscript in preparation (2024)

Optical conductivity of $SrVO_3$ Fermi liquid



Manuscript in preparation (2024)

Optical conductivity of $SrVO_3$ Fermi liquid

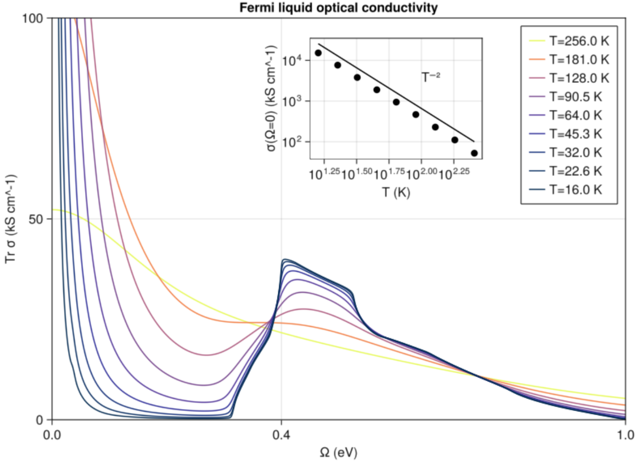


Manuscript in preparation (2024)

Conductivity interpolation

Adaptive frequency interpolation

- Construct smooth representation of data
- Efficiently samples at Chebyshev nodes
- Calculation took one week on 128 threads
- `HChebInterp.jl`



Manuscript in preparation (2024)

Julia HPC* performance lessons

- The garbage collector is single-threaded
- Pre-allocate memory as much as possible (API design)
- Multi-threading performance is sensitive to thread pinning
- `using ThreadPinning; pinthreads(:cores)`
- All other general Julia advice applies, e.g. type-stability

* I've only done shared-memory, CPU programming

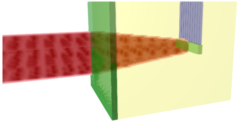
Conclusion

I would like to thank my mentors, Sophie Beck (CCQ) and Jason Kaye (CCQ+CCM), for their ongoing support of this project.

Links:

- Repository: `https://github.com/lxvm/AutoBZCore.jl`
- Paper: SciPost Phys. 15, 062 (2023)

Upcoming subMIT work on photonics inverse design:



FLATIRON INSTITUTE
Center for Computational Quantum Physics

Optical conductivity definitions

Fermi function

$$f(\omega) = 1/(\exp(\beta\omega) + 1)$$

Velocity

$$v_\alpha(\mathbf{k}) = \partial_{k_\alpha} H(\mathbf{k}) + i[H(\mathbf{k}), A_\alpha(\mathbf{k})]$$

where $A_\alpha(\mathbf{k})$ is the Berry connection

Local spectral function

$$A(\mathbf{k}, \omega) = -\operatorname{Im}[(\omega - H(\mathbf{k}) - \Sigma(\omega))^{-1}]/\pi$$

Metasurface design