Jastrow factors in quantum chemistry: the transcorrelated method

Pablo López Ríos

Max-Plack Institute for Solid State Research, Stuttgart

TREX workshop on quantum chemical methods for strongly correlated systems
Institute of Physics, Łódź University of Technology
18–20 April 2023

Full configuration interaction

- The exact solution to the Schrödinger equation in a given orbital basis is:

Full CI wave function

$$\left|\Psi_{\mathrm{FCI}}\right\rangle = \sum_{I \in \mathscr{H}} c_I \left|D_I\right\rangle$$

- Hilbert spaces get intractably large very quickly → approximate solvers can obtain:

Truncated CI wave function

$$\left|\Psi_{\mathrm{CI}}\right\rangle = \sum_{I}^{n_{\mathrm{CI}}} c_I \left|D_I\right\rangle$$

- The truncated CI expansion is inefficient at describing dynamical correlation

Cusp conditions

- Exact wave function Ψ must yield non-divergent local energy
- Kinetic energy must cancel out divergence in Coulomb potential energy at coalescence points, so:

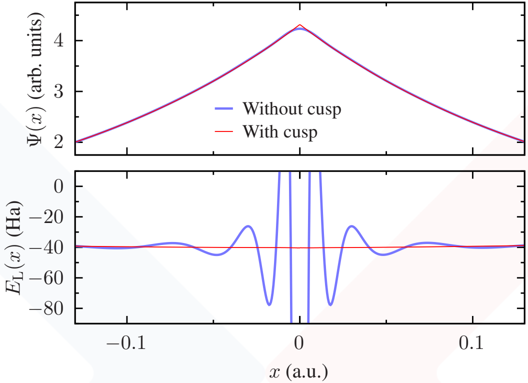
Kato cusp condition

$$\left(\frac{1}{\Psi} \frac{\partial \hat{\Psi}}{\partial r_{ij}} \right)_{r_{ij} \to 0} = \frac{2 q_i q_j \mu_{ij}}{d \pm 1}$$

- Lack of explicit r_{ij} dependence in CI wave function → hard to describe electron–electron cusps
- Electron–nucleus cusps are not satisfied with Gaussian-type orbitals

Cusp conditions

Example of an electron–nucleus cusp:



Carbon atom with cc-pVDZ basis, single Slater determinant

Variational and diffusion quantum Monte Carlo

- VMC and DMC can use any trial wave function; Slater-Jastrow form is standard:

$$\Psi_{\mathrm{SJ}}(\mathbf{R}) = e^{J(\mathbf{R})} D_0(\mathbf{R})$$

where:

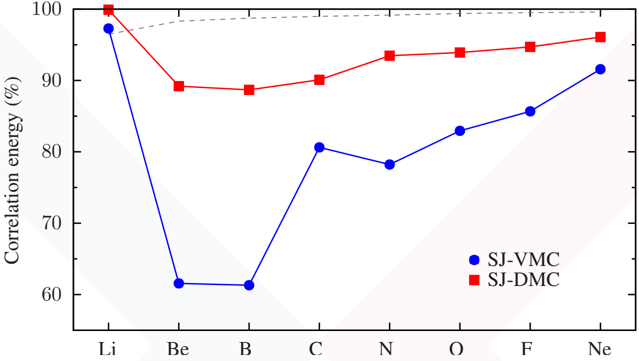
$$J(\mathbf{R}) = \sum_{i<j}^{N_\mathrm{e}} u(r_{ij}) + \sum_{i}^{N_\mathrm{e}} \sum_{I}^{N_\mathrm{n}} \chi(r_{iI}) + \sum_{i<j}^{N_\mathrm{e}} \sum_{I}^{N_\mathrm{n}} f(r_{ij}, r_{iI}, r_{jI}) + \ldots$$

is a Jastrow factor containing e–e, e–n, e–e–n, . . . , terms with parameters:

  - Some parameters are constrained to satisfy Kato cusp conditions, etc
  - Other parameters can be optimized in a VMC framework

Real-space quantum Monte Carlo

Using Slater-Jastrow wave function on first-row atoms:



Real-space quantum Monte Carlo

- Real-space methods:
  - describe dynamical correlation well, even subtle van der Waals effects, and produce accurate total energies
  - describe static correlation inconsistently, resulting in subpar energy differences
- Second-quantized methods:
  - are slow to recover dynamical correlation and yield poor total energies
  - excel at error cancellation and offer accurate energy differences
- There are various ways in which first- and second-quantized approaches can be combined to compound each other's strengths. A very promising one is **transcorrelation**

Transcorrelation

- A Jastrow factor can be used in second quantization if absorbed into Hamiltonian:

$$\hat{H}\left[e^{\hat{J}}|\Psi_{\mathrm{CI}}\rangle\right] = E\left[e^{\hat{J}}|\Psi_{\mathrm{CI}}\rangle\right]$$
$$\left[e^{-\hat{J}}\hat{H}e^{\hat{J}}\right]|\Psi_{\mathrm{CI}}\rangle = E|\Psi_{\mathrm{CI}}\rangle$$
$$\hat{H}_{\mathrm{TC}}|\Psi_{\mathrm{CI}}\rangle = E|\Psi_{\mathrm{CI}}\rangle$$

where the similarity-transformed Hamiltonian is:

$$\hat{H}_{\mathrm{TC}} = e^{-\hat{J}}\hat{H}e^{\hat{J}} = \hat{H} + [\hat{H},\hat{J}] + \frac{1}{2}[[\hat{H},\hat{J}],\hat{J}]$$

- This commutator expansion truncates exactly at second order if J contains up to two-electron terms only, $J = \sum_{i<j} u_{ij}$

Transcorrelated Hamiltonian

$\hat{H}_{\text{TC}}$ is non-Hermitian, connects triple excitations:

Molecular Hamiltonian

$$\hat{H} = \sum_{pq\sigma} h_q^p a_{p\sigma}^\dagger a_{p\sigma} + \frac{1}{2} \sum_{pqrs} V_{rs}^{pq} \sum_{\sigma\tau} a_{p\sigma}^\dagger a_{q\tau}^\dagger a_{s\tau} a_{r\sigma}$$

Transcorrelated Hamiltonian

$$\hat{H}_{\text{TC}} = \sum_{pq\sigma} h_q^p a_{p\sigma}^\dagger a_{p\sigma} + \frac{1}{2} \sum_{pqrs} (V_{rs}^{pq} - K_{rs}^{pq}) \sum_{\sigma\tau} a_{p\sigma}^\dagger a_{q\tau}^\dagger a_{s\tau} a_{r\sigma} - \frac{1}{6} \sum_{pqrstu} L_{stu}^{pqr} \sum_{\sigma\tau\lambda} a_{p\sigma}^\dagger a_{q\tau}^\dagger a_{r\lambda}^\dagger a_{u\lambda} a_{t\tau} a_{s\sigma}$$

Matrix elements

Matrix elements involving u to be obtained by grid integration:

$$
\begin{aligned}
h_q^p &= -\frac{1}{2}\langle \phi_p | \nabla^2 | \phi_q \rangle \\
V_{rs}^{pq} &= \langle \phi_p \phi_q | r_{12}^{-1} | \phi_r \phi_s \rangle \\
K_{rs}^{pq} &= \langle \phi_p \phi_q | \frac{1}{2} \left(\nabla_1^2 u_{12} + \nabla_2^2 u_{12} + (\nabla_1 u_{12})^2 + (\nabla_2 u_{12})^2 \right) \\
&\quad + (\nabla_1 u_{12} \cdot \nabla_1) + (\nabla_2 u_{12} \cdot \nabla_2) | \phi_r \phi_s \rangle \\
L_{stu}^{pqr} &= \langle \phi_p \phi_q \phi_r | \nabla_1 u_{12} \cdot \nabla_1 u_{13} + \nabla_2 u_{21} \cdot \nabla_2 u_{23} \\
&\quad + \nabla_3 u_{31} \cdot \nabla_3 u_{32} | \phi_s \phi_t \phi_u \rangle
\end{aligned}
$$

- $\nabla^2 u$ can be converted to ∇u with integration by parts → only need gradients of J
- K_{rs}^{pq} and L_{stu}^{pqr} evaluation is embarrassingly parallel
- Storage for L_{stu}^{pqr}: $\sim$ 170-orbital basis needs 2TiB memory

TCHInt

We have developed **TCHInt** to handle integral evaluation

- Uses **PySCF** for orbital evaluation

  Orbitals to be provided via `input.molden` file

- Uses **CASINO** Jastrow factor implementation

  Optimized Jastrow factor to be provided via `parameters.casl` file

- Integrals performed on a **grid**

  Uses Treutler-Ahlrichs atom-centred grids from PySCF with adjustable fineness `grid_lvl = 0..9`

- High performance **and** feature rich

  Full MPI parallelization | Exploits fast `dgemm` | Stand-alone binary / on-the-fly library | Mean-field approach

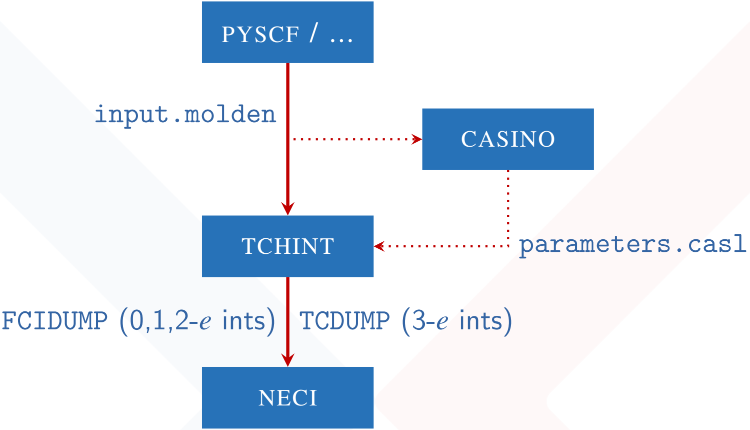
- TCHInt available **on request** for now

  Will be made public in the near future

  TCHINT is planned to become more self-contained

Documentation: https://www2.fkf.mpg.de/alavi/tchint/stable/

TCHInt+NECI workflow for TC-FCIQMC



PYSCF / ...

`input.molden`

CASINO

`parameters.casl`

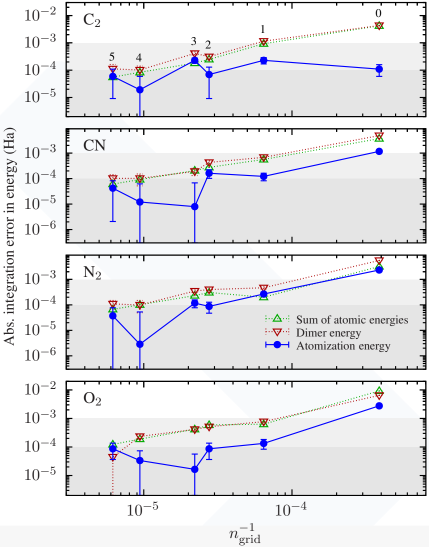
TCHINT

`FCIDUMP` (0,1,2-e ints)

`TCDUMP` (3-e ints)

NECI

Grid integration errors



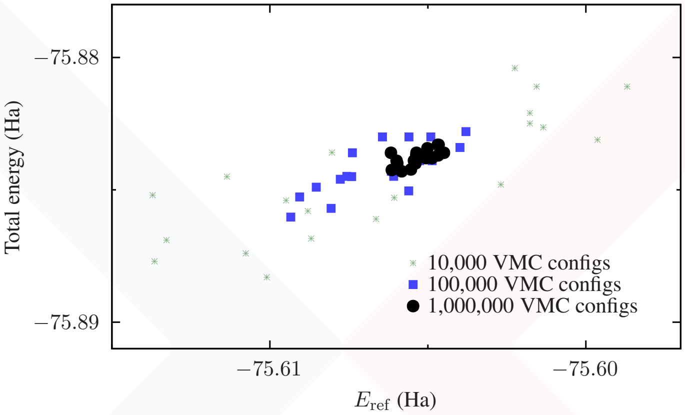
With 0.1 mHa tolerance:

→ `grid_lvl=2` for relative energies

→ `grid_lvl=4` for absolute energies

Handling optimization noise

Choose VMC sample size considering stochastic noise:



Optimization method for TC calculations

One option besides traditional VMC optimization methods:

- Minimize the VMC energy

  Direct application of variational principle

- Minimize the variance of the VMC energy σ^2_{VMC}

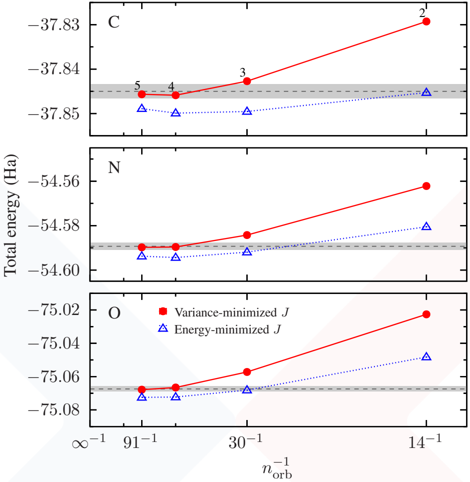
  Since eigenfunctions of $\hat{H}$ yield constant local energy

- Minimize the variance of the reference energy σ^2_{ref}

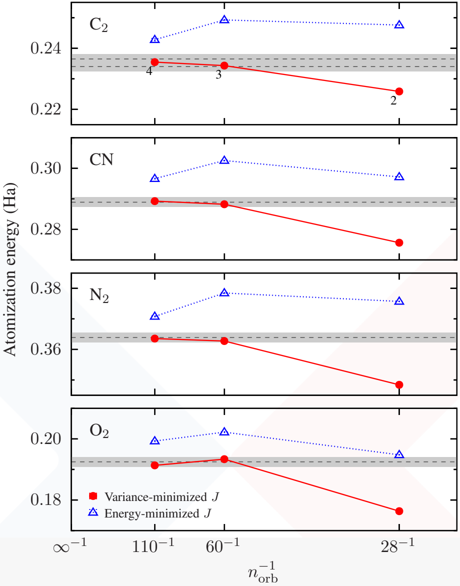
$$\sigma^2_{\mathrm{ref}} = \sum_{I \neq 0} |\langle D_I | \hat{H}_{\mathrm{TC}} | D_0 \rangle|^2$$

→ minimize connectivity of $|D_0\rangle$ → maximize compactness of CI solution

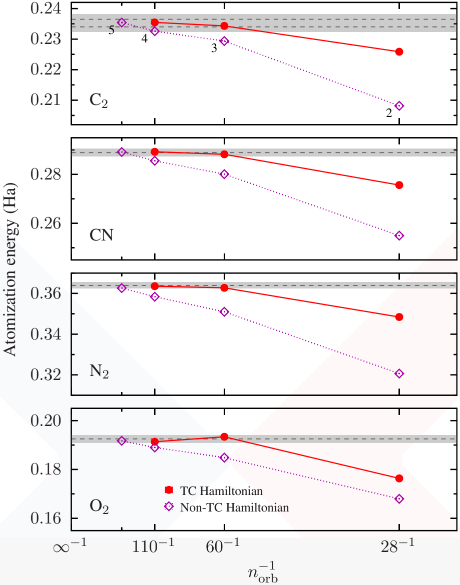
Energy vs variance minimization for E_{tot} of atoms



Energy vs variance minimization for E_{at} of dimers



TC vs non-TC FCI energies



Bibliography

- Papers:
    - T. Kato, cusp conditions (1957)
    - S.F. Boys and N.C. Handy, transcorrelation (1969)
    - N.C. Handy, transcorrelation (1969)
    - H. Luo and A. Alavi, TC-FCIQMC on electron gas (2018)
    - A.J. Cohen et al., TC-FCIQMC on molecules (2019)
    - J.P. Haupt et al., Jastrow optimization for TC (2023) [preprint]
    - E.M. Christlmaier et al., mean-field TC (2023) [preprint]
    - K. Guther et al., NECI code (2020)
    - Q. Sun et al., PySCF code (2020)
    - R.J. Needs et al., CASINO code (2020)
    - P. López Ríos et al., CASINO Jastrow factor (2012)
- Codes:
    - NECI public git repository
    - PySCF public git repository
    - TCHINT documentation
    - CASINO website