European Research Council
Established by the European Commission

Laboratoire de Chimie et Physique Quantiques

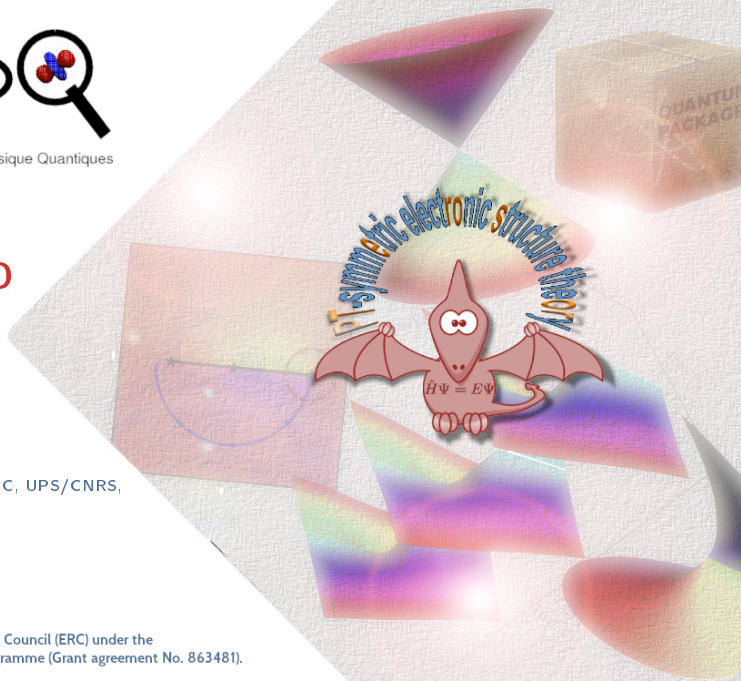
Orbital optimized pCCD solutions for H_4

Fábris Kossoski

29/01/2021

Laboratoire de Chimie et Physique Quantiques, IRSAMC, UPS/CNRS, Toulouse

https://lcpq.github.io/pterosor



PTEROSOR has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (Grant agreement No. 863481).

Usual exponential ansatz:

$$|\Psi\rangle = e^{T}|0\rangle$$

where the excitation operator

$$T = \sum_{ia} t_i^a P_a^\dagger P_i$$

and singlet paired operators

$$P_q^\dagger = c_{q\alpha}^\dagger c_{q\beta}^\dagger$$

Substitution into the Schroedinger equation leads to

$$E = \langle 0|e^{-T} H e^{T}|0\rangle$$
$$0 = \langle 0|P_i^\dagger P_a e^{-T} H e^{T}|0\rangle$$

Equations for the energy

$$E = \langle 0|H|0\rangle + \sum_{ia} t_i^a v_{aa}^{ii}$$

and $k = n_O \times n_V$ polynomial equations for the t-amplitudes:

$$\begin{aligned} 0 =& v_{ii}^{aa} + 2\left(f_a^a - f_i^i - \sum_j v_{aa}^{jj} t_j^a - \sum_b v_{bb}^{ii} t_i^b\right) t_i^a \\ & - 2\left(2v_{ia}^{ia} - v_{ai}^{ia} - v_{aa}^{ii} t_i^a\right) t_i^a \\ & + \sum_b v_{bb}^{aa} t_i^b + \sum_j v_{ii}^{jj} t_j^a + \sum_{jb} v_{bb}^{jj} t_j^a t_i^b \end{aligned}$$

where f_q^p is an element of the Fock operator and $v_{rs}^{pq} = \langle \phi_p \phi_q | V_{ee} | \phi_r \phi_s \rangle$ is a two-electron integral.

Since we want to optimize the orbitals, the energy must be expressed as an expectation value, and thus we have to introduce the de-excitation operator:

$$Z = \sum_{ia} z_a^i P_i^\dagger P_a$$

Writing $\varepsilon = \langle 0|(1+Z)e^{-T}He^{T}|0\rangle$ leads to $k = n_O \times n_V$ polynomial equations for the z-amplitudes:

$$\begin{aligned} 0 =& v_{aa}^{ii} + 2\left(f_a^a - f_i^i - \sum_j v_{aa}^{jj} t_j^a - \sum_b v_{bb}^{ii} t_i^b\right) z_a^i \\ & - 2\left(2v_{ia}^{ia} - v_{ai}^{ai} - v_{aa}^{ii} t_i^a\right) z_a^i - 2v_{aa}^{ii}\left(\sum_j z_a^j t_j^a + \sum_b z_b^i t_i^b\right) \\ & + \sum_b v_{aa}^{bb} z_b^i + \sum_j v_{jj}^{ii} z_a^j + \sum_{jb} t_j^b (v_{bb}^{ii} z_a^j + v_{aa}^{jj} z_b^i) \end{aligned}$$

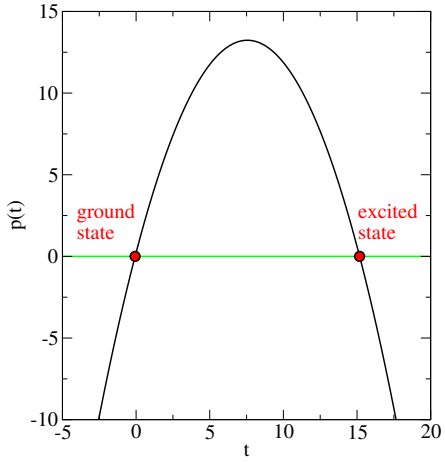


Figure: pCCD polynomial $p(t)$

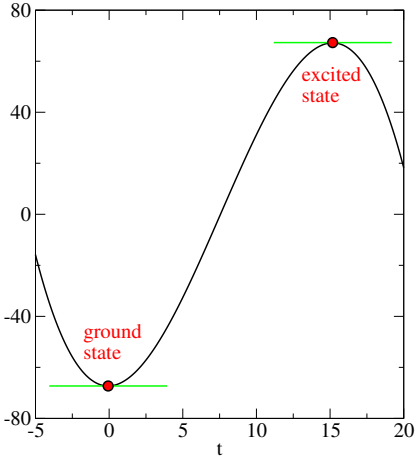


Figure: Integrated pCCD polynomial $\int p(t)dt$

- Multidimensional problem is more complicated:

$$\begin{aligned} p_1(t_1, t_2 \ldots t_k) &= 0 \\ p_2(t_1, t_2 \ldots t_k) &= 0 \\ &\vdots \\ p_k(t_1, t_2 \ldots t_k) &= 0 \end{aligned}$$

- In matrix notation:

$$\mathbf{p(t)} = \mathbf{0} \tag{1}$$

$$\mathbf{p}(\mathbf{t}) = 0 \tag{2}$$

- With the Newton-Raphson root finding method, the solution is found iteratively. Taylor expand $\mathbf{p}(\mathbf{t})$ to first order:

$$\mathbf{p}(\mathbf{t}) \approx \mathbf{p}(\mathbf{t}_0) + \mathbf{J}(\mathbf{t}_0)\Delta\mathbf{t}$$

- where $\Delta\mathbf{t} = \mathbf{t} - \mathbf{t}_0$, and $\mathbf{J} = \partial\mathbf{p}/\partial\mathbf{t}$ is the Jacobian.

$$\mathbf{p}(\mathbf{t}) = 0 \tag{2}$$

- With the Newton-Raphson root finding method, the solution is found iteratively. Taylor expand $\mathbf{p}(\mathbf{t})$ to first order:

$$\mathbf{p}(\mathbf{t}) \approx \mathbf{p}(\mathbf{t}_0) + \mathbf{J}(\mathbf{t}_0)\Delta\mathbf{t}$$

- where $\Delta\mathbf{t} = \mathbf{t} - \mathbf{t}_0$, and $\mathbf{J} = \partial\mathbf{p}/\partial\mathbf{t}$ is the Jacobian.
- In the next step, move to the approximate solution $\mathbf{p}(\mathbf{t}) \approx \mathbf{0}$:

$$0 = \mathbf{p}(\mathbf{t}_0) + \mathbf{J}(\mathbf{t}_0)\Delta\mathbf{t}$$

- Such that:

$$\mathbf{t}_{n+1} = \mathbf{t}_n - [\mathbf{J}(\mathbf{t}_n)]^{-1}\mathbf{g}(\mathbf{t}_n)$$

- This involves building and inverting the $k \times k$ Jacobian matrix

- Newton-Raphson algorithm:

$$\mathbf{t}_{n+1} = \mathbf{t}_n - [\mathbf{J}(\mathbf{t}_n)]^{-1}\mathbf{g}(\mathbf{t}_n)$$

- One usually employs a constant diagonal approximation to the full Jacobian:

$$J_{ia,ia} = \frac{\partial p_i^a(\mathbf{t})}{\partial t_i^a} \approx 2(f_a^a - f_i^i)$$

- This is good enough for the ground state but not for excited states

- Newton-Raphson algorithm:

$$\mathbf{t}_{n+1} = \mathbf{t}_n - [\mathbf{J}(\mathbf{t}_n)]^{-1}\mathbf{g}(\mathbf{t}_n)$$

- One usually employs a constant diagonal approximation to the full Jacobian:

$$J_{ia,ia} = \frac{\partial p_i^a(\mathbf{t})}{\partial t_i^a} \approx 2(f_a^a - f_i^i)$$

- This is good enough for the ground state but not for excited states
- For that, the full Jacobian is required
- Empolying the complete (t-dependent) diagonal works most of the time:

$$J_{ia,ia} = 2f_a^a - 2f_i^i - 4v_{ia}^{ia} + 2v_{ai}^{ia} + v_{aa}^{aa} + v_{ii}^{ii} - \sum_j v_{jj}^{aa} t_j^a - \sum_b v_{ii}^{bb} t_i^b$$

- Newton-Raphson algorithm:

$$\mathbf{t}_{n+1} = \mathbf{t}_n - [\mathbf{J}(\mathbf{t}_n)]^{-1}\mathbf{g}(\mathbf{t}_n)$$

- One usually employs a constant diagonal approximation to the full Jacobian:

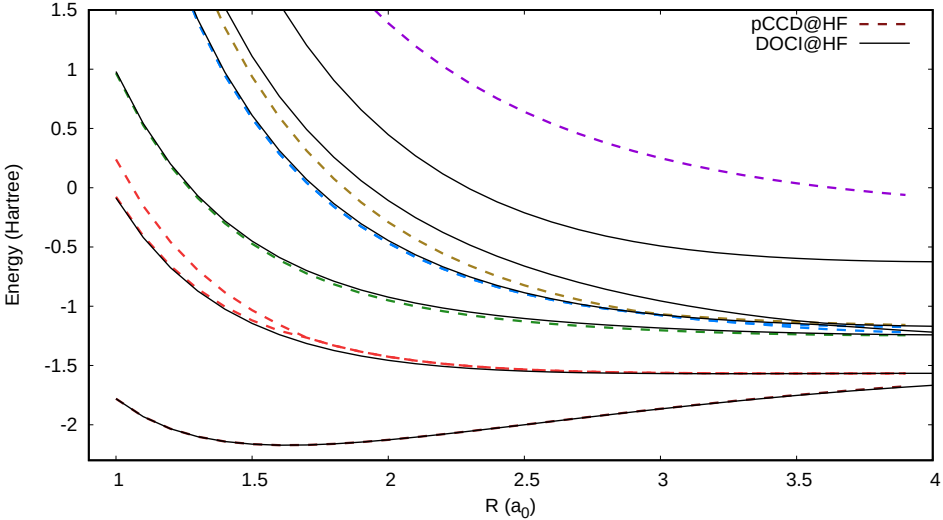
$$J_{ia,ia} = \frac{\partial p_i^a(\mathbf{t})}{\partial t_i^a} \approx 2(f_a^a - f_i^i)$$

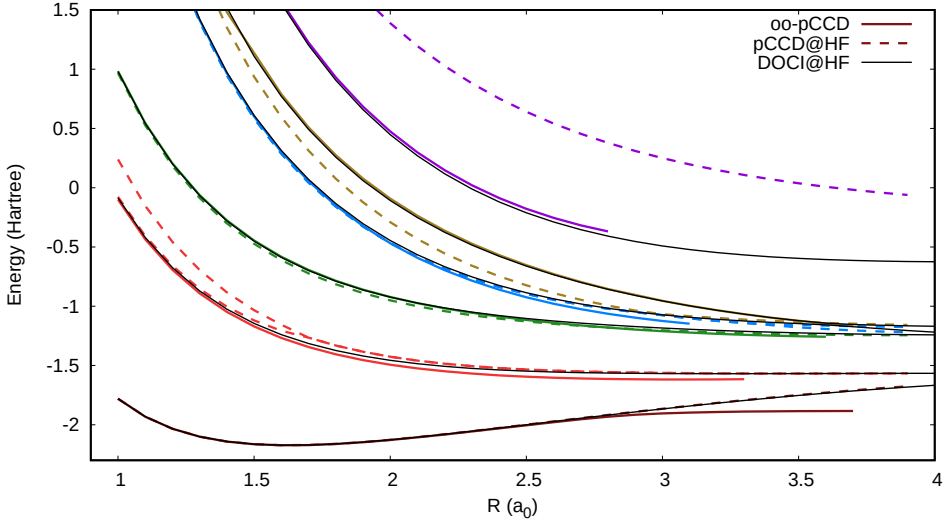
- This is good enough for the ground state but not for excited states
- For that, the full Jacobian is required
- Empolying the complete (t-dependent) diagonal works most of the time:

$$J_{ia,ia} = 2f_a^a - 2f_i^i - 4v_{ia}^{ia} + 2v_{ai}^{ia} + v_{aa}^{aa} + v_{ii}^{ii} - \sum_j v_{jj}^{aa} t_j^a - \sum_b v_{ii}^{bb} t_i^b$$

- Equation for the z-amplitudes are linear. Newton-Raphson is exact!

H_4, STO-6G


pCCD@HF
DOCI@HF
Energy (Hartree)
R (a_0)
1.5
1
0.5
0
-0.5
-1
-1.5
-2
1
1.5
2
2.5
3
3.5
4


oo-pCCD
pCCD@HF
DOCI@HF
Energy (Hartree)
R (a_0)
1.5
1
0.5
0
-0.5
-1
-1.5
-2
1
1.5
2
2.5
3
3.5
4