Development of wave-function based methods

Qauntum Package 2.0

Diata Traore

Laboratoire de Chimie Théorique - UMR7616
Sorbonne Université

Slides and materials on dtraore97.github.io/ressources/QP

Table of Contents

1. Introduction
2. Available electronic structure methods
3. Quantum Package

Quantum Package - purposes

Source code : quantumpackage.github.io

What for ?

*"Quantum Package is an open-source programming environment for quantum chemistry specially designed for **wave-function methods**. Its main goal is the **development** of determinant-driven **selected Configuration Interaction (sCI) methods** and multireference second-order perturbation theory (PT2)."*
(J. Chem. Theory Comput. 2019, 15, 6, 3591–3609)

Quantum Package - purposes

Source code : quantumpackage.github.io

What for ?

"Quantum Package is an open-source programming environment for quantum chemistry specially designed for ***wave-function methods****. Its main goal is the* ***development*** *of determinant-driven* ***selected Configuration Interaction (sCI) methods*** *and multireference second-order perturbation theory (PT2)."*
(J. Chem. Theory Comput. 2019, 15, 6, 3591–3609)

This presentation:

1. Reminder on electronic structure methods: HF, CISD, CIPSI.
2. Intro to Quantum Package.

General introduction about quantum chemistry

The many-body Schrödinger equation

$$\hat{H}\Psi(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N) = E\Psi(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N)$$

where

$$\hat{H} = -\frac{1}{2}\sum_i \nabla_i^2 + \frac{1}{2}\sum_{i \neq j} \frac{1}{|\vec{r}_i - \vec{r}_j|} + \sum_i v(\vec{r}_i)$$

General introduction about quantum chemistry

The many-body Schrödinger equation

$$\hat{H}\Psi(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N) = E\Psi(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N)$$

where

$$\hat{H} = -\frac{1}{2}\sum_i \nabla_i^2 + \frac{1}{2}\sum_{i\neq j}\frac{1}{|\vec{r}_i - \vec{r}_j|} + \sum_i v(\vec{r}_i)$$

Strategies:

- Hartree-Fock : one determinant, averaged electronic interaction.
- Configuration interaction : multi-determinant, use actual Coulomb interaction.
- Selected Configuration interaction : select the more important determinants.
- ...

Hartree-Fock approach

Hartree-Fock approximation :
A Slater determinant :

$$\Psi^{\mathrm{HF}}(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_{\mathrm{N}}) = (N!)^{-1/2} \begin{vmatrix} \phi_i(\mathbf{x}_1) & \phi_j(\mathbf{x}_1) & \dots & \phi_k(\mathbf{x}_1) \\ \phi_i(\mathbf{x}_2) & \phi_j(\mathbf{x}_2) & \dots & \phi_k(\mathbf{x}_1) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_i(\mathbf{x}_{\mathrm{N}}) & \phi_j(\mathbf{x}_{\mathrm{N}}) & \dots & \phi_k(\mathbf{x}_{\mathrm{N}}) \end{vmatrix}$$

where

$$\phi_i(\mathbf{r}, \omega) = \alpha(\omega) \sum_{\mu} c_{\mu i} \chi_{\mu}(\vec{r})$$

$$\phi_i(\mathbf{r}, \omega) = \beta(\omega) \underbrace{\sum_{\mu} c_{\mu i} \chi_{\mu}(\vec{r})}_{\psi_i(\mathbf{r})}$$

with Gaussian-type basis functions :

$$\chi_{\mu}(\vec{r}) = r^{l_{\mu}} \left(\sum_{i} d_{\mu,i} e^{-\alpha_{\mu} r^2} \right) \mathcal{Y}_{l_{\mu}, m_{\mu}}(\theta, \phi)$$

Hartree-Fock approach

Hartree-Fock approximation :
A Slater determinant :

$$\Psi^{\mathrm{HF}}(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_{\mathrm{N}}) = (N!)^{-1/2} \begin{vmatrix} \phi_i(\mathbf{x}_1) & \phi_j(\mathbf{x}_1) & \cdots & \phi_k(\mathbf{x}_1) \\ \phi_i(\mathbf{x}_2) & \phi_j(\mathbf{x}_2) & \cdots & \phi_k(\mathbf{x}_1) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_i(\mathbf{x}_{\mathrm{N}}) & \phi_j(\mathbf{x}_{\mathrm{N}}) & \cdots & \phi_k(\mathbf{x}_{\mathrm{N}}) \end{vmatrix}$$

Hartree-Fock equation :

$$E_0^{\mathrm{HF}} = \min_{\Psi^{\mathrm{HF}}} \langle \Psi^{\mathrm{HF}} | \hat{H} | \Psi^{\mathrm{HF}} \rangle$$

Euler-Lagrange with the constraint $\langle \phi_i | \phi_j \rangle = \delta_{ij}$:

$$\hat{f} \phi_i(\mathbf{x}) = \epsilon_i \phi_j(\mathbf{x})$$

Hartree-Fock estimation of the ground state energy :

$$\underbrace{E^{\mathrm{correlation}}}_{\text{☹}} = E^{\mathrm{exact}} - E^{\mathrm{HF}}$$

Hartree-Fock approach

Hartree-Fock approximation :
A Slater determinant :

$$\Psi^{\mathrm{HF}}(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_{\mathrm{N}}) = (N!)^{-1/2} \begin{vmatrix} \phi_i(\mathbf{x}_1) & \phi_j(\mathbf{x}_1) & \ldots & \phi_k(\mathbf{x}_1) \\ \phi_i(\mathbf{x}_2) & \phi_j(\mathbf{x}_2) & \ldots & \phi_k(\mathbf{x}_1) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_i(\mathbf{x}_{\mathrm{N}}) & \phi_j(\mathbf{x}_{\mathrm{N}}) & \ldots & \phi_k(\mathbf{x}_{\mathrm{N}}) \end{vmatrix}$$

$\longrightarrow$ **Configuration interaction** methods

FCI approach

The full configuration interaction (FCI) wave function :

$$|\Psi^{\mathrm{FCI}}\rangle = c_{\mathrm{HF}}|\psi^{\mathrm{HF}}\rangle + \sum_{ar} c_a^r|\psi_a^r\rangle + \sum_{a>b,\, r<s} c_{ab}^{rs}|\psi_{ab}^{rs}\rangle + \sum_{a<b<c,\, r<s<t} c_{abc}^{rst}|\psi_{abc}^{rst}\rangle + \dots$$

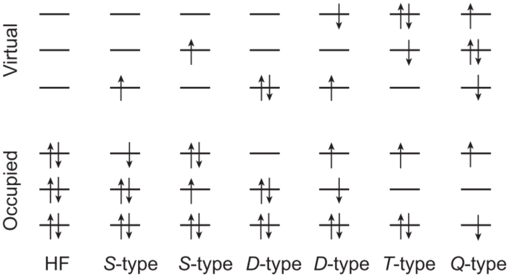

FIGURE 3.18 Excited Slater determinants generated from a ground state HF reference configuration.

(V.P. Gupta, Principles and applications of quantum chemistry, Elsevier (2016))

Approximations of the FCI

The full configuration interaction (FCI) wave function :

$$|\Psi^{\mathrm{FCI}}\rangle = c_{\mathrm{HF}}|\psi^{\mathrm{HF}}\rangle + \sum_{ar} c_a^r|\psi_a^r\rangle + \sum_{a>b,\, r<s} c_{ab}^{rs}|\psi_{ab}^{rs}\rangle + \sum_{a<b<c,\, r<s<t} c_{abc}^{rst}|\psi_{abc}^{rst}\rangle + \dots$$

Truncation of the CI space :

- CIS (configuration interaction with simple excitations)
- CISD (configuration interaction with simple and double excitations)

Approximations of the FCI

The full configuration interaction (FCI) wave function :

$$|\Psi^{\text{FCI}}\rangle = c_{\text{HF}}|\psi^{\text{HF}}\rangle + \sum_{ar} c_a^r|\psi_a^r\rangle + \sum_{a>b,\, r<s} c_{ab}^{rs}|\psi_{ab}^{rs}\rangle + \sum_{a<b<c,\, r<s<t} c_{abc}^{rst}|\psi_{abc}^{rst}\rangle + \dots$$

Truncation of the CI space :

- CIS (configuration interaction with simple excitations)
- CISD (configuration interaction with simple and double excitations)
- ☞ **CIPSI** : **C**onfiguration **I**nteraction **P**erturbatively **S**elected **I**teratively

(B. Huron, J. P. Malrieu, and P. Rancurel, J. Chem. Phys. ***58****, 5745, (1973))*
(E. Giner, A. Scemama, M. Caffarel, Canadian J. Chem., ***91****, 879-885 (2013))*

(B. Huron, J. P. Malrieu, and P. Rancurel, J. Chem. Phys. ***58****, 5745, (1973))*
(E. Giner, A. Scemama, M. Caffarel, Canadian J. Chem., ***91****, 879-885 (2013))*

Configuration-Interaction Perturbatively Selected Iteratively algorithm

Step 1 :

$$|\Psi^{(k)}\rangle = \sum_{I \in \mathcal{R}^{(k)}} c_I^{(k)} |I\rangle$$

*(B. Huron, J. P. Malrieu, and P. Rancurel, J. Chem. Phys. **58**, 5745, (1973))*
*(E. Giner, A. Scemama, M. Caffarel, Canadian J. Chem., **91**, 879-885 (2013))*

Configuration-Interaction Perturbatively Selected Iteratively algorithm

Step 1 :

$$|\Psi^{(k)}\rangle = \sum_{I\in\mathcal{R}^{(k)}} c_I^{(k)} |I\rangle$$

Step 2 :

$$\epsilon_{P,\mathrm{PT2}}^{(k)} = -\frac{|\langle\Psi^{(k)}|\hat{H}|P\rangle|^2}{\langle P|\hat{H}|P\rangle - \langle\Psi^{(k)}|\hat{H}|\Psi^{(k)}\rangle}$$

$\rightarrow$ Selection of the largest contributions $|\epsilon_{P,\mathrm{PT2}}^{(k)}|$.

(B. Huron, J. P. Malrieu, and P. Rancurel, J. Chem. Phys. ***58****, 5745, (1973))*
(E. Giner, A. Scemama, M. Caffarel, Canadian J. Chem., ***91****, 879-885 (2013))*

Configuration-Interaction Perturbatively Selected Iteratively algorithm

Step 1 :

$$|\Psi^{(k)}\rangle = \sum_{I \in \mathcal{R}^{(k)}} c_I^{(k)} |I\rangle$$

Step 2 :

$$\epsilon_{P,\mathrm{PT2}}^{(k)} = -\frac{|\langle \Psi^{(k)} | \hat{H} | P \rangle|^2}{\langle P | \hat{H} | P \rangle - \langle \Psi^{(k)} | \hat{H} | \Psi^{(k)} \rangle}$$

→ Selection of the largest contributions $|\epsilon_{P,\mathrm{PT2}}^{(k)}|$.

Step 3 :

$$\text{If } |\sum_P \epsilon_{P,\mathrm{PT2}}^{(k)}| > \text{threshold} \rightarrow \text{Step 1}$$

Manual

https://quantum-package.readthedocs.io/en/master/

- Installation guide
- Index of programs and commands (users and programmers oriented)

Launch a calculation

(Link to the Quick-start guide)

Step 1: Create an xyz file

`Number of atom`	`3`
`(Blank or comment line)`	`HCN molecule`
`Atom X Y Z` (In angström !)	`H 0. 0. 0.`
`Atom X Y Z`	`C 0. 0. 1.064`
`Atom X Y Z`	`N 0. 0. -1.156`

Launch a calculation

(Link to the Quick-start guide)

Step 1: Create an xyz file

```
Number of atom                                      3
(Blank or comment line)                             HCN molecule
Atom X Y Z (In angström !)                          H 0.  0.  0.
Atom X Y Z                                          C 0.  0.  1.064
Atom X Y Z                                          N 0.  0.  -1.156
```

Step 2: Create the EZFIO database

```
qp create_ezfio - b "6-31g" -o hcn hcn.xyz
```

The list of available basis is available in the data/basis folder.

Launch a calculation

(Link to the Quick-start guide)

Step 1: Create an xyz file

`Number of atom`	`3`
`(Blank or comment line)`	`HCN molecule`
`Atom X Y Z` (In angström !)	`H 0. 0. 0.`
`Atom X Y Z`	`C 0. 0. 1.064`
`Atom X Y Z`	`N 0. 0. -1.156`

Step 2: Create the EZFIO database

```
qp create_ezfio - b "6-31g" -o hcn hcn.xyz
```

The list of available basis is available in the data/basis folder.

Step 3: Run the program

`qp run scf` ⟶ Self-consistent field Hartree-Fock calc.

`qp run fci` ⟶ CIPSI calc.

The index of programs is available on the QP website.

Create a plugin

Step 1: From the root, activate the QP shell mode

```
${QP_ROOT}/bin/qpsh
```

Create a plugin

Step 1: From the root, activate the QP shell mode

```
${QP_ROOT}/bin/qpsh
```

Step 2: Once in the `plugins` folder, type

```
qp_plugins create -n plugin_name [-r plugin_repo] [list of needed modules]
```

Create a plugin

Step 1: From the root, activate the QP shell mode

```
${QP_ROOT}/bin/qpsh
```

Step 2: Once in the `plugins` folder, type

```
qp_plugins create -n plugin_name [-r plugin_repo] [list of needed modules]
```

Step 3: Install the plugin

```
qp_plugins install plugin_name
```

Create a plugin

Step 1: From the root, activate the QP shell mode

```
${QP_ROOT}/bin/qpsh
```

Step 2: Once in the `plugins` folder, type

```
qp_plugins create -n plugin_name [-r plugin_repo] [list of needed modules]
```

Step 3: Install the plugin

```
qp_plugins install plugin_name
```

Remarks:

In the created folder,

- `NEED` file : you can add the list of needed modules
- `README.rst` file : to document your module
- `plugin_name.irp.f` : default first script file

What can you code ?

programs

```
program program_label
 implicit none
 BEGIN_DOC
 !  Documentation
 END_DOC
 print *, "Hello world"
end
```

(call via `qp run program_label`)

What can you code ?

programs

```
program program_label
 implicit none
 BEGIN_DOC
 !  Documentation
 END_DOC
 print *, "Hello world"
end
```

(call via `qp run program_label`)

routines

```
subroutine routine_label
 implicit none
 BEGIN_DOC
 !  Documentation
 END_DOC
 integer ::  i
 double precision ::  var
end
```

What can you code ?

functions

```
double precision function_label([var1, var2])
 implicit none
 BEGIN_DOC
 END_DOC
 integer ::  i
 double precision, intent(in) ::  var1, var2
end
```

What can you code ?

functions

```
double precision function_label([var1, var2])
 implicit none
 BEGIN_DOC
 END_DOC
 integer ::  i
 double precision, intent(in) ::  var1, var2
end
```

providers

```
BEGIN_PROVIDER [double precision, var1]]
&BEGIN_PROVIDER [integer, table1, (size1, size2)]]
 implicit none
 BEGIN_DOC
 END_DOC
 integer ::  i
END_PROVIDER
```

Compilation

⟶ To compile the plugin, type `ninja` ⟵

↺

Debug (read the error message, the online guide, ...)

Tutorial

- **Option 1** : Launch computation of Hartree-Fock and CIS calculations and exploit output to understand the wave-function.
- **Option 2** : Create a module with a program to compute the electronic density on a grid.

Ressources

Introduction to quantum chemistry

- *Modern Quantum Chemistry - Introduction to Advanced Electronic Structure Theory*, Attila Szabo and Neil S. Ostlund (Dover)
- *Molecular Electronic-Structure Theory*, Trygve Helgaker, Poul Jorgensen, Jeppe Olsen (Wiley)
- *Introduction to computational chemistry*, Frank Jensen (Wiley)
- Julien Toulouse lecture notes: Link

Quantum Package, IRPF90

- QP webpage: Link
- Github repo: Link
- Reference for QP: J. Chem. Theory Comput. 2019, 15, 6, 3591–3609
- Anthony Scemama webpage about IRPF90: Link