EMSL

NWChem: Hartree-Fock, Density Functional Theory

Daniel Mejia-Rodriguez,

Edoardo Aprà & Niri Govind

2023 CyberWorkshop

Hartree-Fock

- Functionality
- Input
- Wavefunctions
- Initial MO vectors
- Direct and semidirect algorithms
- Convergence, files, and restarting

Hartree-Fock Functionality

- Energies and gradients
  - Closed-shell (RHF)
  - Spin-restricted, high-spin open-shell (ROHF)
  - Spin-unrestricted open-shell (UHF)
- Analytic second derivatives (RHF and UHF)
- Finite point groups
- Will be used as first step in all correlated methods (e.g. MP2, CC, etc …)
- Resolution of the identity (energy)

Input

- SCF input block, e.g.,

```
 scf
    triplet
     uhf
end
```

- Defaults
  - Restricted-spin wavefunction (ROHF)
  - Accuracy suitable for non-floppy molecule geometry optimization
  - Symmetry as defined in the geometry

CH_2
molecule

$^3B_1$ CH_2 ROHF and UHF optimizations

```
geometry units au
  C   0   0       0
  H   0   1.868 -0.818
  symmetry c2v
end

basis
  H library 3-21g; C library 3-21g
end

scf; sym off; triplet; end  #default is ROHF
task scf optimize

scf; uhf; end
task scf optimize
```

Density-Functional Theory

- Functionality
- Input
- XC functionals
- Grid & Convergence options

DFT Functionality in a nutshell

- *Gaussian function*-based DFT (aka LCAO)
  - *energies*,
  - *gradients* and
  - *second derivatives*
- finite *symmetry*

- Exchange-Correlation functionals for
  - Closed-Shell systems and Open-Shell systems

Exchange-Correlation Functionals

- Hartree-Fock Exchange
- Traditional functionals: Density & density gradient
    - LDA, BP, BLYP, PBE, PW91,…
- Hybrid functionals: Inclusion of HF exchange
    - B3LYP, PBE0, BeckeHalfandHalf,…
- Meta functionals: Inclusion of kinetic energy
    - TPSS, PKZB, Minnesota functionals,…
- Range-separated functionals
    - CAM-B3LYP, LC-PBE0,…
- DFT + empirical dispersion (DFT+ D)
    - Based on Grimme’s implementation
- Double Hybrid functionals: DFT + MP2
    - Based on Grimme’s implementation

Input

- DFT input block, e.g.,

```
dft
 mult 1
end
```

- Defaults (similar to Hartree-Fock)
  - Local density approximation (**LDA**)
  - Accuracy suitable for non-floppy molecule geometry optimization
  - Symmetry as defined in the geometry

Open Shell Input

- DFT input block, e.g.,

```
dft
  mult 3
end
```

- Unrestriced Open Shell Default (different from Hartree-Fock)
- RODFT is available

Minimal Input Example

- Minimal input (all defaults)

```
geometry; ne 0 0 0; end
basis; ne library cc-pvdz; end
task dft
```

- Performs a closed-shell N^4 DFT calculation using the local density approximation on the neon atom (no fitting)

Simple DFT Input Example

- Input with default DFT input (single point LDA calculation)

```
echo  # echoes the input in the output file
start silane   # name of files
title silane   # title of the calculation in output
charge 0.
geometry
  si         0.00000000     0.00000000     0.00000000
  h          0.75252170    -0.75252170     0.75252170
  h         -0.75252170     0.75252170     0.75252170
  h          0.75252170     0.75252170    -0.75252170
  h         -0.75252170    -0.75252170    -0.75252170
end
dft; mult 1;end
basis
  * library cc-pvdz
end
task dft   # specifies the task → energy by default
```

- EMSL Basis Set Exchange: **https://bse.pnl.gov/bse/portal**

Changing the exchange-correlation

```
echo
start silane
title silane
geometry
      si          0.00000000     0.00000000     0.00000000
      h           0.75252170    -0.75252170     0.75252170
      h          -0.75252170     0.75252170     0.75252170
      h           0.75252170     0.75252170    -0.75252170
      h          -0.75252170    -0.75252170    -0.75252170
end

basis
  * library cc-pvdz
end

dft
  xc b3lyp # B3LYP
end

task dft
```

```
dft
  xc becke88 lyp #BLYP
end
```

```
dft
  xc becke88 perdew86
end
```

…

Many other combinations possible…

Important DFT keywords

xc: controls the choice of the exchange-correlation

convergence: controls the convergence (energy, density…)

grid: specifies the grid

mult: specifies the multiplicity

odft: specify open shell calculation (redundant when **mult** is there)

iterations: controls the number of iterations

smear: useful for degenerate states

```
SINGLET
dft
 grid fine
 convergence energy 1e-08
 xc b3lyp  #B3LYP
 mult 1
end
```

```
TRIPLET
dft
  odft
  grid fine
  convergence energy 1e-08
  xc b3lyp  #B3LYP
  mult 3
end
```

Putting it all together

```
echo
start silane
title silane

geometry
      si         0.00000000     0.00000000     0.00000000
      h          0.75252170    -0.75252170     0.75252170
      h         -0.75252170     0.75252170     0.75252170
      h          0.75252170     0.75252170    -0.75252170
      h         -0.75252170    -0.75252170    -0.75252170
end

basis
  * library cc-pvdz
end

dft
  grid fine
  convergence energy 1e-08
  xc b3lyp # B3LYP
  mult 1
end
task dft
```

Geometry Optimization

```
echo
start silane

geometry
      si           0.00000000     0.00000000     0.00000000
      h            0.75252170    -0.75252170     0.75252170
      h           -0.75252170     0.75252170     0.75252170
      h            0.75252170     0.75252170    -0.75252170
      h           -0.75252170    -0.75252170    -0.75252170
end

basis
  * library cc-pvdz
end

dft
  grid xfine
  convergence energy 1e-08
  xc b3lyp # B3LYP
  mult 1
end

task dft optimize
```

Frequencies

```
start silane

geometry
  si          0.00000000     0.00000000     0.00000000
  h           0.75252170    -0.75252170     0.75252170
  h          -0.75252170     0.75252170     0.75252170
  h           0.75252170     0.75252170    -0.75252170
  h          -0.75252170    -0.75252170    -0.75252170
end

basis
  * library cc-pvdz
end

dft
  grid xfine
  convergence energy 1e-08
  xc b3lyp # B3LYP
  mult 1
end

task dft frequencies
```

Combining Calculations I

```
start silane

geometry
    si         0.00000000     0.00000000     0.00000000
    h          0.75252170    -0.75252170     0.75252170
    h         -0.75252170     0.75252170     0.75252170
    h          0.75252170     0.75252170    -0.75252170
    h         -0.75252170    -0.75252170    -0.75252170
end

basis
  * library cc-pvdz
end

dft
  grid xfine
  convergence energy 1e-08
  xc b3lyp # B3LYP
  mult 1
end
task dft optimize
task dft frequencies
```

Combining Calculations II

```
…

dft
   xc b3lyp  #B3LYP
   mult 1
end

task dft optimize
task dft frequencies

dft
   odft
   xc becke88 lyp #BLYP
   mult 3
end

task dft optimize
```

Restarting Calculations

```
echo
restart silane

geometry
      si          0.00000000     0.00000000     0.00000000
      h           0.75252170    -0.75252170     0.75252170
      h          -0.75252170     0.75252170     0.75252170
      h           0.75252170     0.75252170    -0.75252170
      h          -0.75252170    -0.75252170    -0.75252170
end

basis
  * library cc-pvdz
end

dft
  grid xfine
  convergence energy 1e-08
  xc b3lyp # B3LYP
  mult 1
end

task dft
```

Restart files
- `silane.db`
- `silane.movecs`

Using Old Molecular Orbitals

```
echo
start silane

geometry
      si         0.00000000     0.00000000     0.00000000
      h          0.75252170    -0.75252170     0.75252170
      h         -0.75252170     0.75252170     0.75252170
      h          0.75252170     0.75252170    -0.75252170
      h         -0.75252170    -0.75252170    -0.75252170
end

basis
  * library cc-pvdz
end

dft
  grid xfine
  convergence energy 1e-08
  xc b3lyp # B3LYP
  mult 1
  vectors input old.movecs output b3lyp.movecs
end
task dft
```

Organizing Your Files

```
echo
start silane

permanent_dir /home/yourname/silane/b3lyp
scratch_dir /scratch

geometry
      si          0.00000000     0.00000000     0.00000000
      h           0.75252170    -0.75252170     0.75252170
      h          -0.75252170     0.75252170     0.75252170
      h           0.75252170     0.75252170    -0.75252170
      h          -0.75252170    -0.75252170    -0.75252170
end

basis
  * library cc-pvdz
end
dft
  grid xfine
  convergence energy 1e-08
  xc b3lyp  #B3LYP
  mult 1
end
task dft optimize
```

Customizing The Basis

```
...
geometry
     si          0.00000000     0.00000000     0.00000000
     h1          0.75252170    -0.75252170     0.75252170
     h2         -0.75252170     0.75252170     0.75252170
     h3          0.75252170     0.75252170    -0.75252170
     h4         -0.75252170    -0.75252170    -0.75252170
end

basis
      si library 6-31G
      h1 library h sto-3g
      h2 library h 6-31g
      h3 library h 3-21g
      h4 library h 6-31g*
end

...
```

Including empirical dispersion in DFT

```
...
geometry
      ...
end

basis
      ...
end
dft
 xc b3lyp
 disp vdw 4
end
task dft optimize
```

S. Grimme J. Comp. Chem. 25 1463 (2004)
S. Grimme J. Comp. Chem. 271787 (2006)

Semi-empirical hybrid DFT + MP2 Double Hybrid Functionals

```
...
geometry
      ...
end

basis
      ...
end
dft
  xc HFexch 0.53 becke88 0.47 lyp 0.73 mp2 0.27
  dftmp2 direct
  direct
  convergence energy 1e-8
  iterations 100
end
```

S. Grimme, J. Chem. Phys., 124, 034108 (2006)

Other Capabilities

- Charge density fitting (Dunlap scheme)
  - 4-center, 2-electron Coulomb integrals → 3-center integrals (N^3)
  - Very fast for traditional DFT (pure density based functionals, no HF Exchange)
  - Cheaper and better parallel scaling

- Direct or on-the-fly evaluation of integrals
  - All integrals evaluated as needed
  - Useful for large systems on large numbers of processors

- Effective Core Potentials

Charge-Density Fitting

- Important difference between DFT and SCF
  - **Additional fitting basis set** (reduces cost from N^4 --> N^3)

```
geometry; ne 0 0 0; end

basis "ao basis" spherical
 ne library def2-tzvp
end

basis "cd basis"
 ne library "Weigend Coulomb Fitting"
end

task dft
```

Effective Core Potentials

- Reduces the cost of calculation for heavy elements
  - Additional input field required to define potential

```
geometry; ne 0 0 0; end

  ecp spherical
    * library Stuttgart_RSC_1997_ECP
  end

  basis "ao basis"
   ni library "Stuttgart_RSC_1997_ECP"
  end

  task dft
```

Grid Options

- Numerical integration keywords and targets using Mura-Knowles radial and Lebedev angular quadratures:

```
dft; grid xcoarse; end  (1d-4 au)
dft; grid coarse; end   (1d-5 au)
dft; grid medium; end   (1d-6 au; default)
dft; grid fine; end     (1d-7 au)
dft; grid xfine; end    (1d-8 au)
dft; grid huge; end
```

- Addition quadrature choices, e.g.,

```
dft; grid eumac medium; end
dft; grid ssf lebedev 75 11; end
```

(= G98 fine)

Modifying Accuracy

- Controlling accuracy
  - **Schwarz screening is invoked for density*integral < 10-accCoul, `accCoul` default = 10**
  - **e.g., `tolerances accCoul 12`**
- When to change it?
  - **Diffuse basis/floppy molecules**
  - **Changing from energy to optimizations, frequencies, etc.**
  - **Don’t forget to increase grid accuracy too!**

Convergence

- *DIIS*, *level-shifting*, and *damping* are available
- Default is DIIS with no damping. Level-shifting is invoked when the HOMO-LUMO gap is less than **`hl_tol`** (default is 0.05 atomic units)
- Control of DIIS, level-shifting, and damping: **`convergence lshift 0.1 damp 40 diis 5`**
- When invoked can be by iteration count

```
convergence ncydp 5
```

- or by change in total energy

```
convergence ncydp 0 dampon 1d6 \
dampoff 1d-2
```

Fractional occupation of MOs

- The `SMEAR` keyword is useful in cases with many degenerate states near the HOMO (e.g., metallic clusters). Molecular Orbitals near the gap will be occupied with a distribution a la Fermi-Dirac corresponding to a finite temperature.

- `SMEAR <real smear default 0.001>`

Questions?

Thank you