HORTON & ChemTools

Farnaz Heidar-Zadeh, Paul Ayers, Toon Verstraelen, Esteban Vohringer, Carlos Cardenas,…

So theory guys have got it made
in rooms free of pollution.
Instead of problems with the reflux,
they have only solutions…
In other words, experimentalists
will likely die of cancer
From working hard, yet fruitlessly,
till theory gives the answer.

Thomas A. Holme

$$ NSERC, Compute Canada; Canada Research Chairs; Marie Curie Actions (EU); FWO; Fondecyt

HORTON & ChemTools

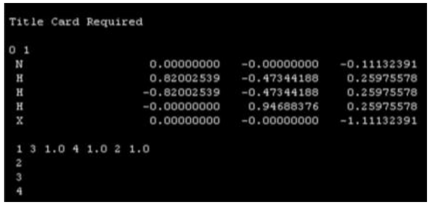
HORTON quantum chemistry package

Helpful **O**pen-source **R**esearch **TO**ol for **N**-electron systems

- **Free** & **open-source** electronic structure code with a strictly **modular design**.
- Motivated by the **difficulty to add new features** to the existing quantum chemistry programs.
- Written primarily in Python 3, with C++/Fortran only where necessary for computational performance.
- Computationally efficient enough to be helpful, without compromising **code-readability** and **user-friendliness**.
- Well-tested, well-documented, and follows modern “best practices” for software engineering.
- Helpful for prototyping, developing, and exploring new methods!
- **quantumelephant.org**

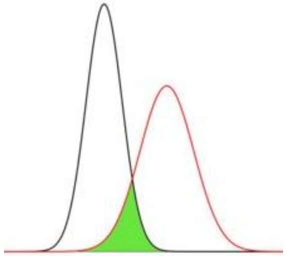
Modules

iodata

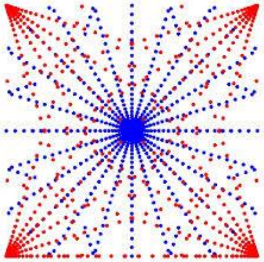


```
Title Card Required

0 1
 N                    0.00000000   -0.00000000   -0.11132391
 H                    0.82002539   -0.47344188    0.25975578
 H                   -0.82002539   -0.47344188    0.25975578
 H                   -0.00000000    0.94688376    0.25975578
 X                    0.00000000   -0.00000000   -1.11132391

 1 3 1.0 4 1.0 2 1.0
 2
 3
 4
```

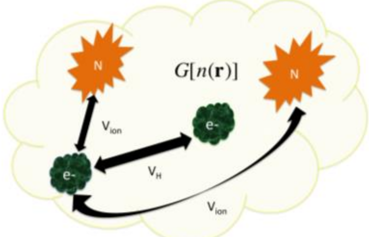
gbasis



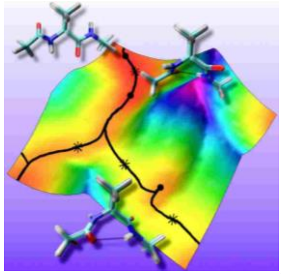
grid



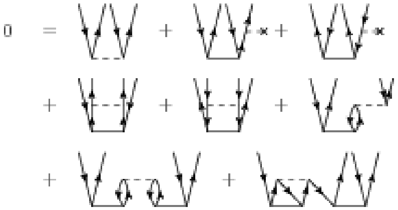
meanfield



gopt



postscf



Gain Chemical Insight from Computations

Farnaz Heidar-Zadeh
Toon Verstraelen
Stijn Fias
J. Contreras-García

Paul Ayers & his group!

S. Liu

R. Miranda-Quintana
G. Sánchez Díaz

M. Franco-Pérez

E. Vöhringer-Martinez
C. Cárdenas

J. Anderson

C. Rong

Greenland
Iceland
Sweden
Norway
United Kingdom
Canada
Poland
Ukraine
Germany
France
Kazakhstan
Mongolia
Italy
Spain
United States
North Atlantic Ocean
Turkey
China
South Korea
Japan
Iraq
Iran
Afghanistan
Pakistan
Algeria
Libya
Egypt
Saudi Arabia
India
Mexico
Thailand
Niger
Chad
Sudan
Venezuela
Nigeria
Ethiopia
Colombia
Kenya
DR Congo
Tanzania
Brazil
Peru
Angola
Bolivia
Namibia
Botswana
South Atlantic
South Africa
Argentina

GAIN CHEMICAL INSIGHT FROM COMPUTATIONS

✧ **Free** & **open-source** Python package for interpreting the results of quantum chemistry calculations!

✧ **Easy-to-use core functionality** to compute fundamental descriptors of conceptual quantum chemistry, together with a **flexible set of utilities to easily test your ideas**!

✧ Readable, well-documented, well-tested & user-friendly.

✧ ChemTools can be used as:

- Python Library
- Python Scripts (command line)

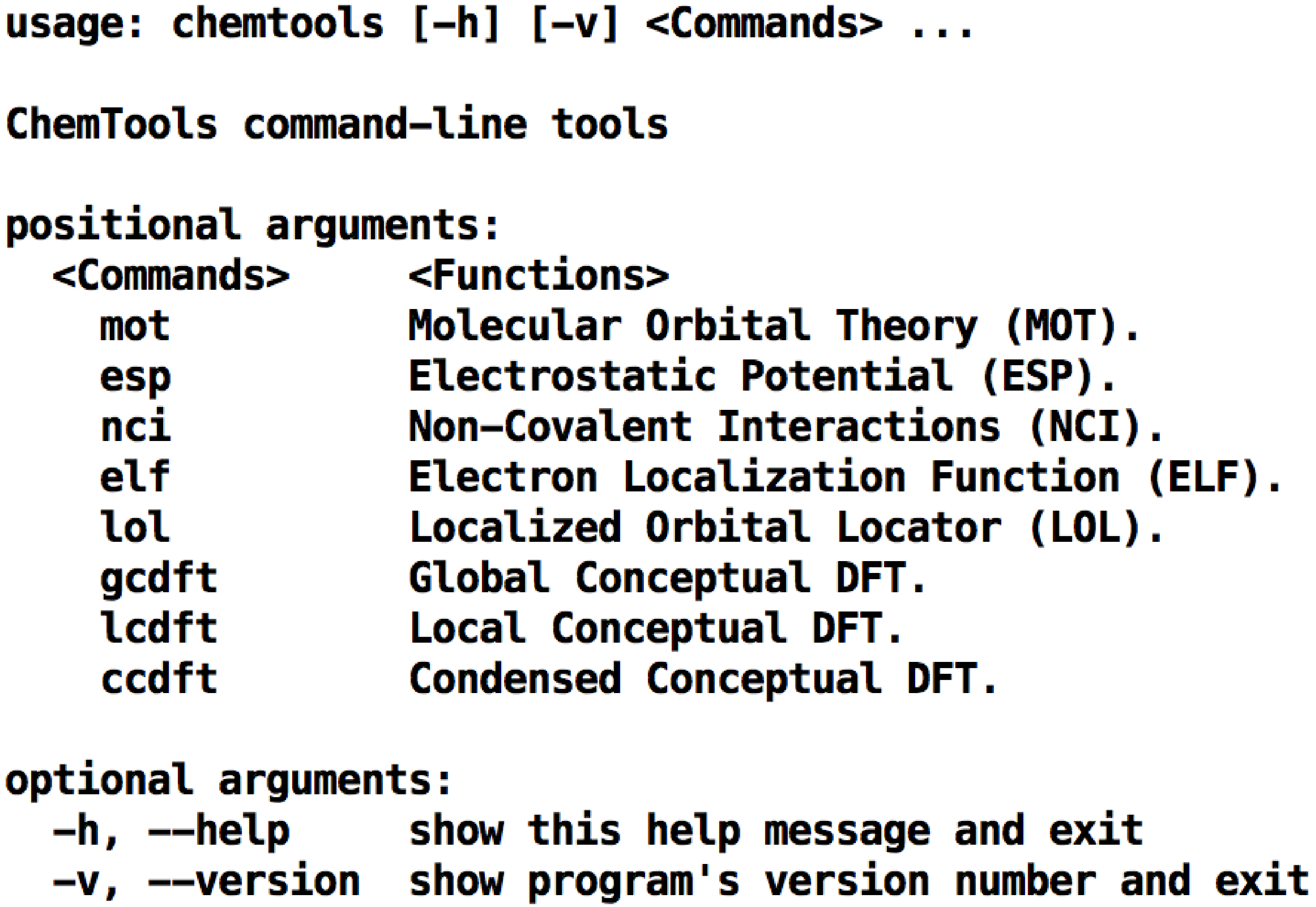
✧ Release Date: June 2019, Visit: **chemtools.org**

✧ Examples and installation instructions (click on "material")

https://wiki.lct.jussieu.fr/workshop/index.php/ChemTools_Workshop_2019

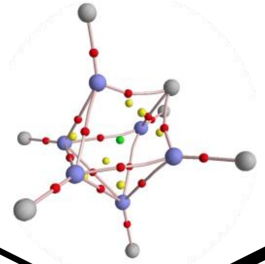
Command-Line Functionality

```
$ chemtools -h
```

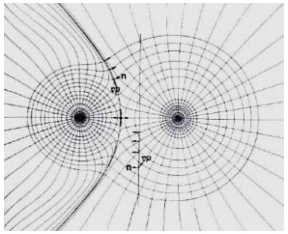


```
usage: chemtools [-h] [-v] <Commands> ...

ChemTools command-line tools

positional arguments:
  <Commands>      <Functions>
    mot           Molecular Orbital Theory (MOT).
    esp           Electrostatic Potential (ESP).
    nci           Non-Covalent Interactions (NCI).
    elf           Electron Localization Function (ELF).
    lol           Localized Orbital Locator (LOL).
    gcdft         Global Conceptual DFT.
    lcdft         Local Conceptual DFT.
    ccdft         Condensed Conceptual DFT.

optional arguments:
  -h, --help      show this help message and exit
  -v, --version   show program's version number and exit
```

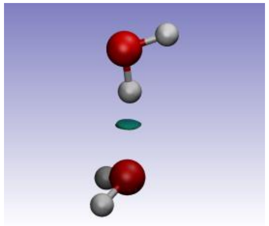
Topological Analysis



AIM Partitioning

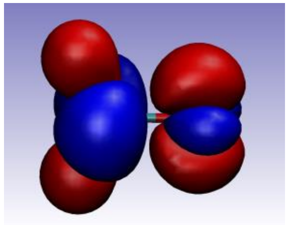


DFT-Based Descriptors

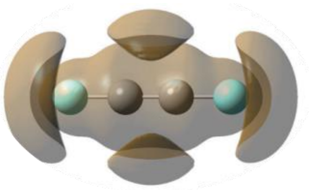


MODULES

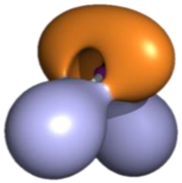
Conceptual DFT



Density Matrix Analysis



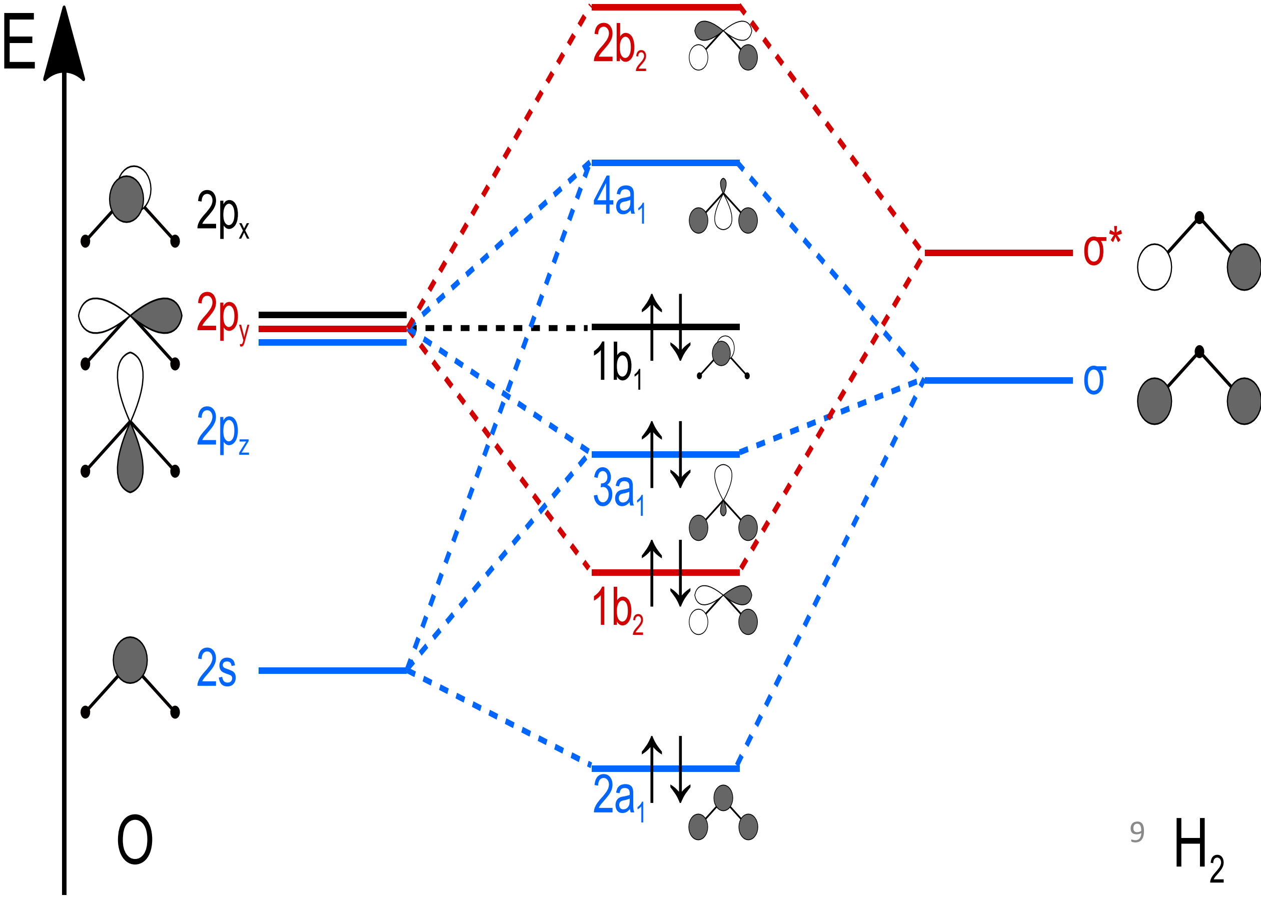
Orbital-Based Descriptors



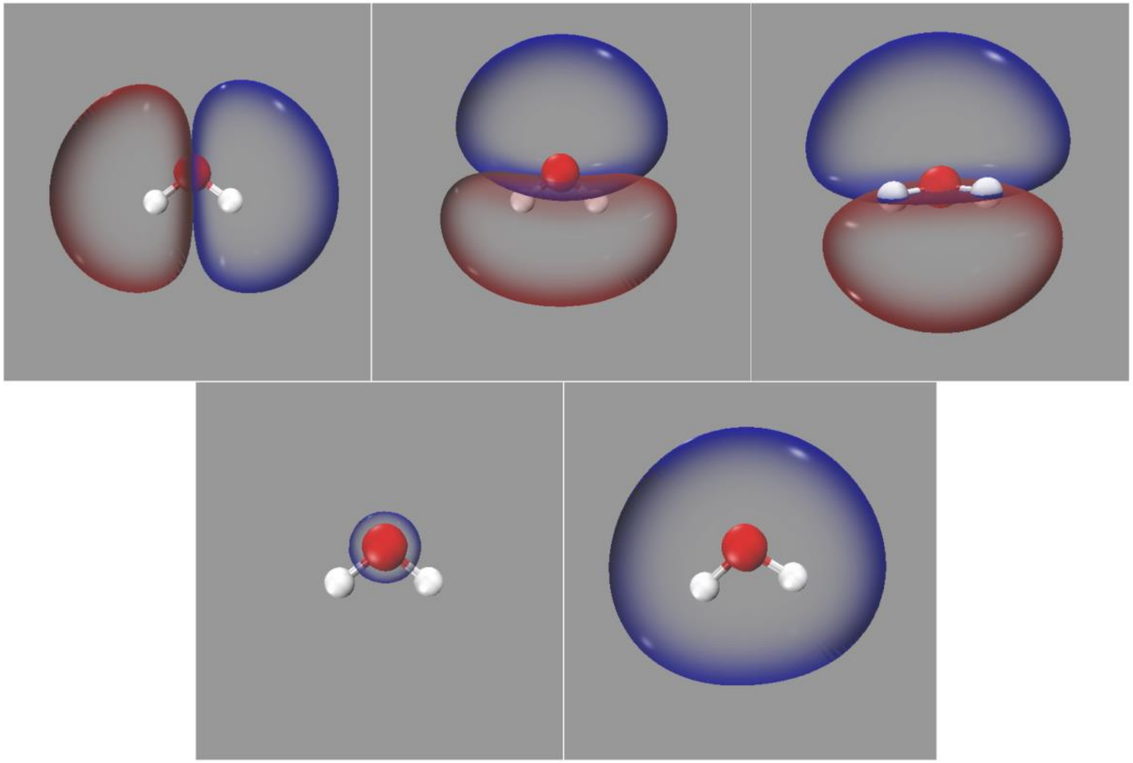
Included & Planned Features

Molecular Orbital Theory (MOT) Descriptors and Analysis

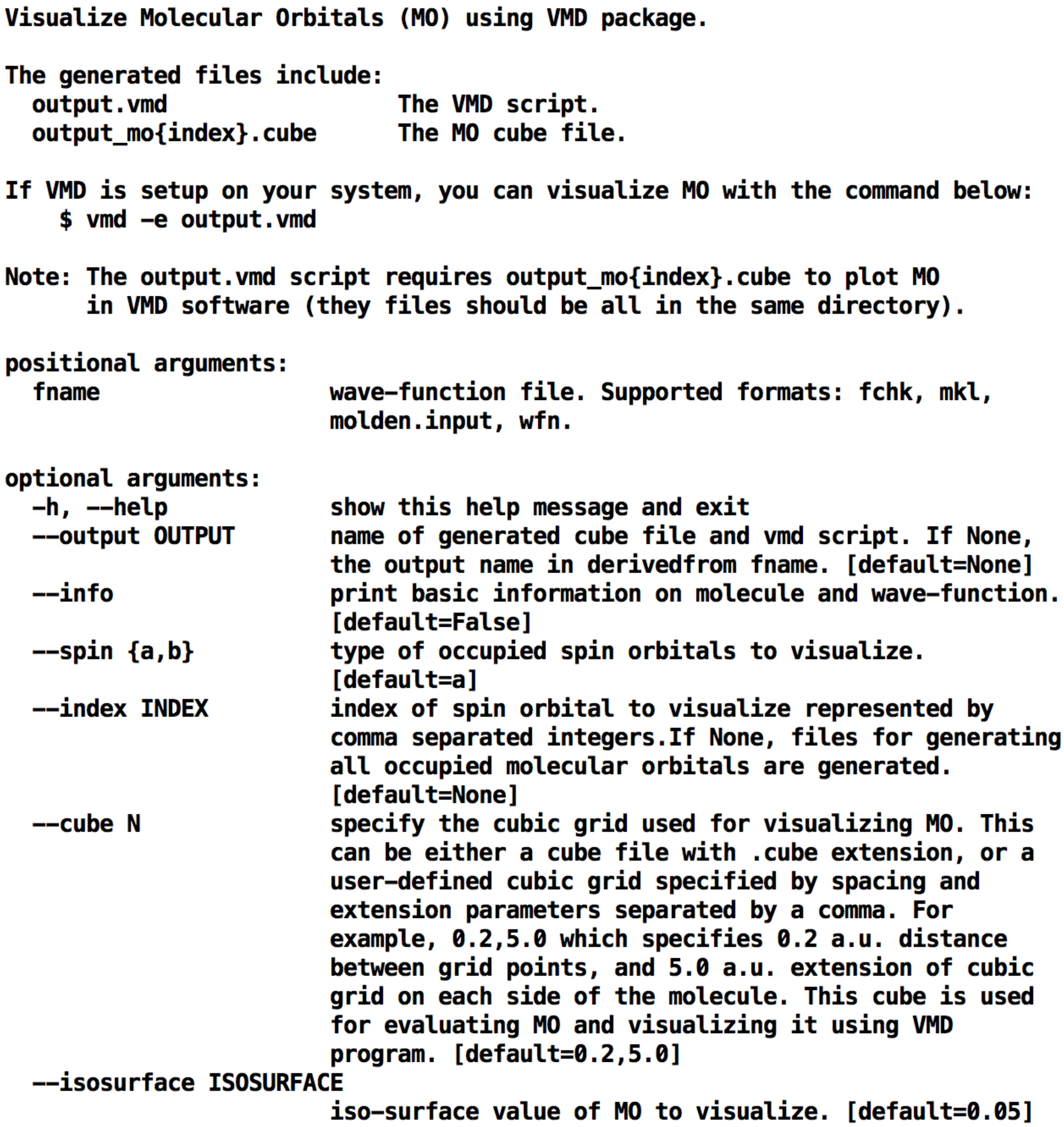
- Orbital Energies; Visualize Orbitals
- Draw MO diagram with AO contributions/energies.
- Mulliken, Lowdin, Ruedenberg, Knizia population/bonding analysis.
- Localized Orbitals
- Oxidation states



Molecular Orbital Theory (MOT)

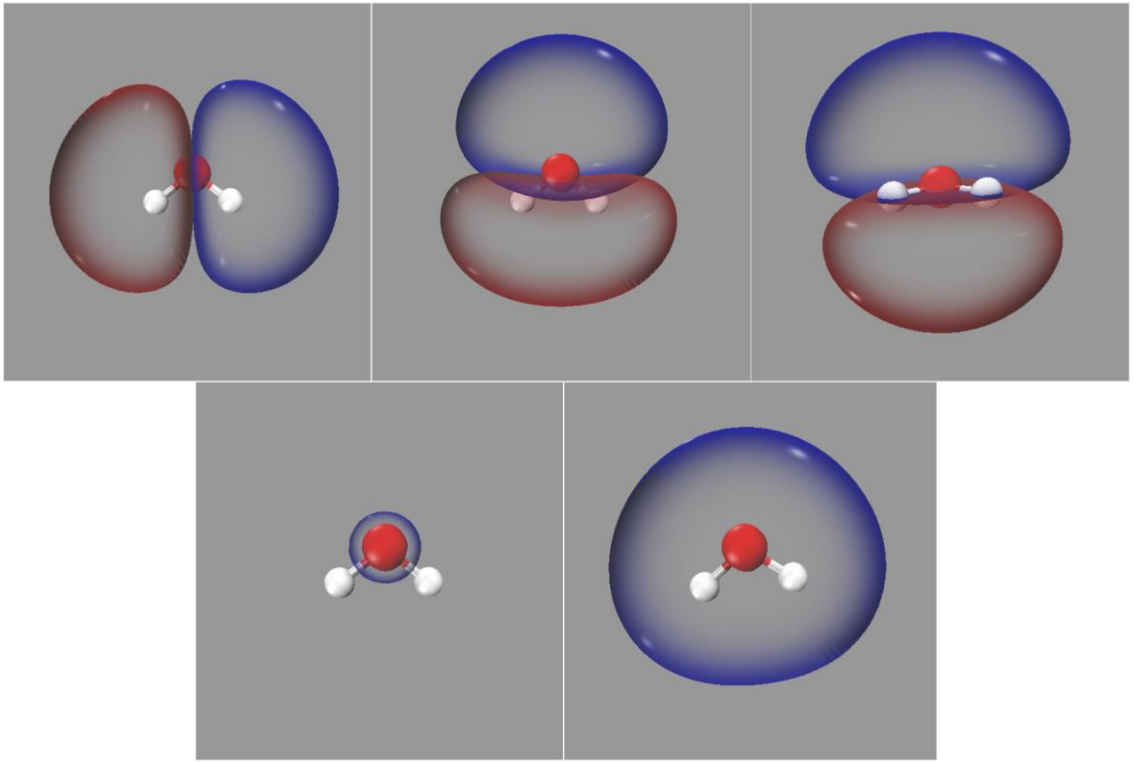


```
$ chemtools mot -h
```



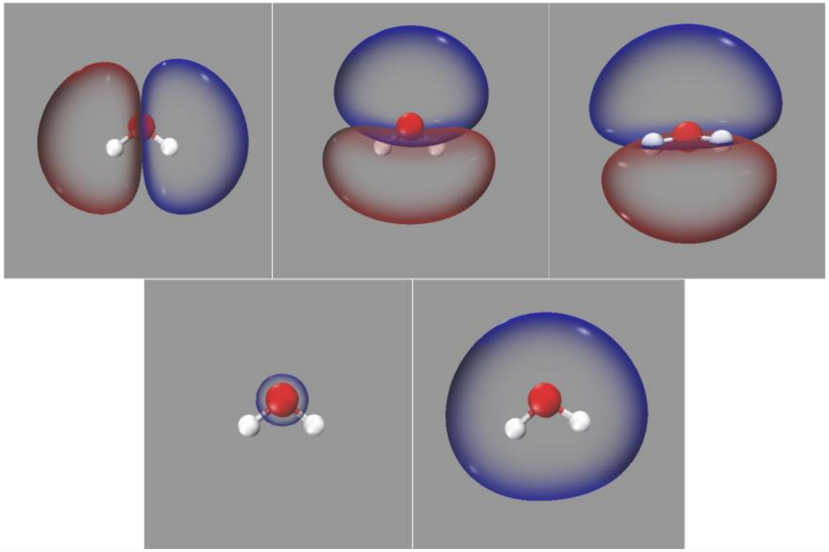
```
Visualize Molecular Orbitals (MO) using VMD package.

The generated files include:
  output.vmd                   The VMD script.
  output_mo{index}.cube        The MO cube file.

If VMD is setup on your system, you can visualize MO with the command below:
    $ vmd -e output.vmd

Note: The output.vmd script requires output_mo{index}.cube to plot MO
      in VMD software (they files should be all in the same directory).

positional arguments:
  fname                 wave-function file. Supported formats: fchk, mkl,
                        molden.input, wfn.

optional arguments:
  -h, --help            show this help message and exit
  --output OUTPUT       name of generated cube file and vmd script. If None,
                        the output name in derivedfrom fname. [default=None]
  --info                print basic information on molecule and wave-function.
                        [default=False]
  --spin {a,b}          type of occupied spin orbitals to visualize.
                        [default=a]
  --index INDEX         index of spin orbital to visualize represented by
                        comma separated integers.If None, files for generating
                        all occupied molecular orbitals are generated.
                        [default=None]
  --cube N              specify the cubic grid used for visualizing MO. This
                        can be either a cube file with .cube extension, or a
                        user-defined cubic grid specified by spacing and
                        extension parameters separated by a comma. For
                        example, 0.2,5.0 which specifies 0.2 a.u. distance
                        between grid points, and 5.0 a.u. extension of cubic
                        grid on each side of the molecule. This cube is used
                        for evaluating MO and visualizing it using VMD
                        program. [default=0.2,5.0]
  --isosurface ISOSURFACE
                        iso-surface value of MO to visualize. [default=0.05]
```

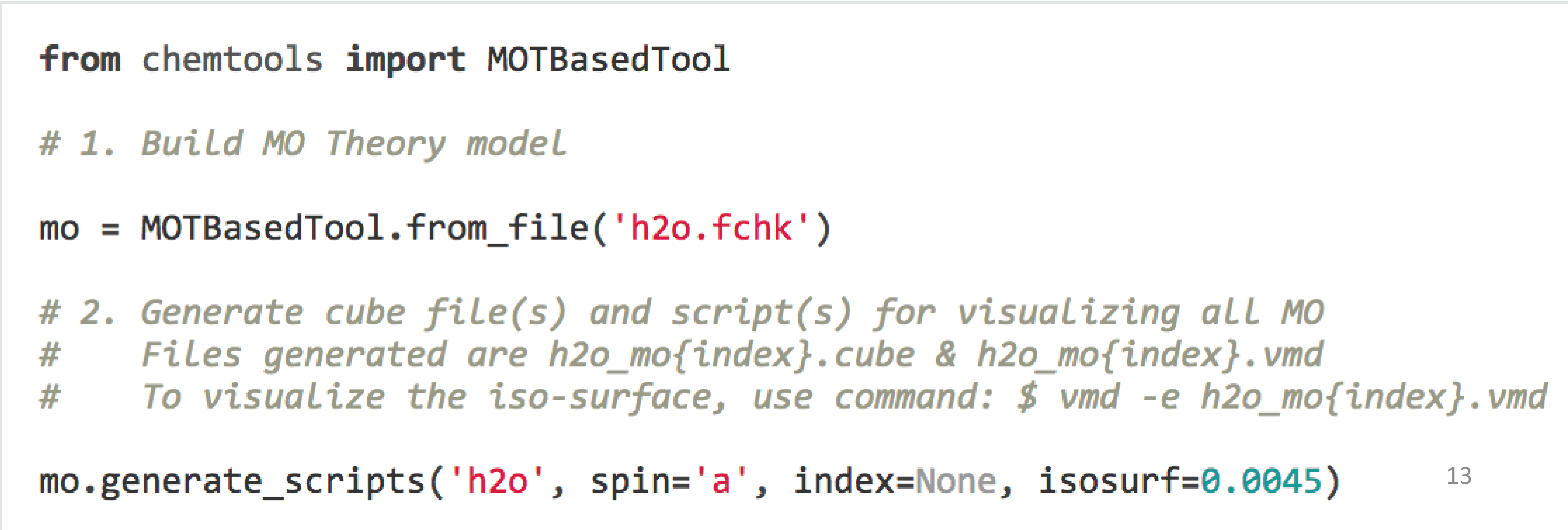
Molecular Orbital Theory (MOT)



```
$ chemtools mot -h

$ chemtools mot h2o.fchk
```

Molecular Orbital Theory (MOT)

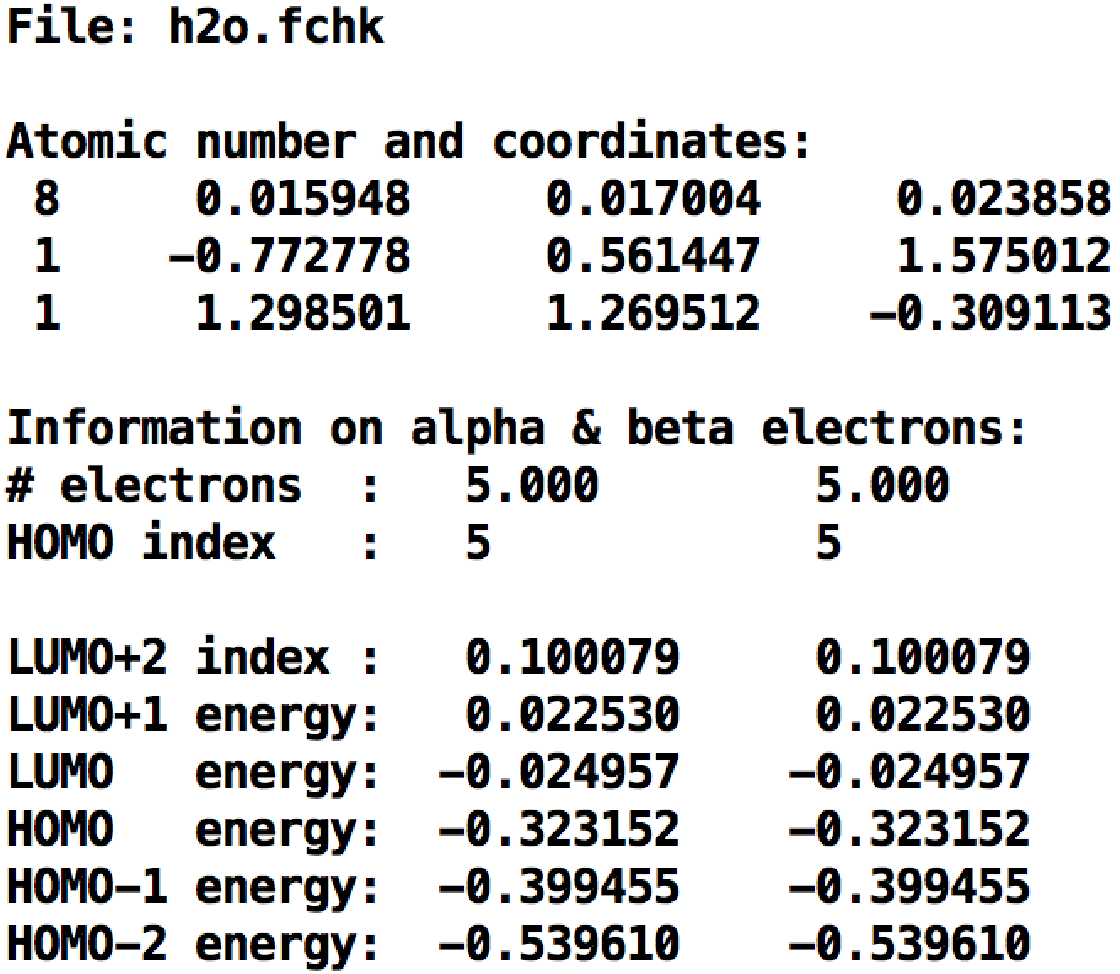




```
from chemtools import MOTBasedTool

# 1. Build MO Theory model

mo = MOTBasedTool.from_file('h2o.fchk')

# 2. Generate cube file(s) and script(s) for visualizing all MO
#    Files generated are h2o_mo{index}.cube & h2o_mo{index}.vmd
#    To visualize the iso-surface, use command: $ vmd -e h2o_mo{index}.vmd

mo.generate_scripts('h2o', spin='a', index=None, isosurf=0.0045)
```

Molecular Orbital Theory (MOT)

```
$ chemtools mot h2o.fchk --info
```



```
File: h2o.fchk

Atomic number and coordinates:
 8      0.015948      0.017004      0.023858
 1     -0.772778      0.561447      1.575012
 1      1.298501      1.269512     -0.309113

Information on alpha & beta electrons:
# electrons  :   5.000         5.000
HOMO index   :   5             5

LUMO+2 index :   0.100079      0.100079
LUMO+1 energy:   0.022530      0.022530
LUMO   energy:  -0.024957     -0.024957
HOMO   energy:  -0.323152     -0.323152
HOMO-1 energy:  -0.399455     -0.399455
HOMO-2 energy:  -0.539610     -0.539610
```

Included & Planned Features

Density-Based Descriptors

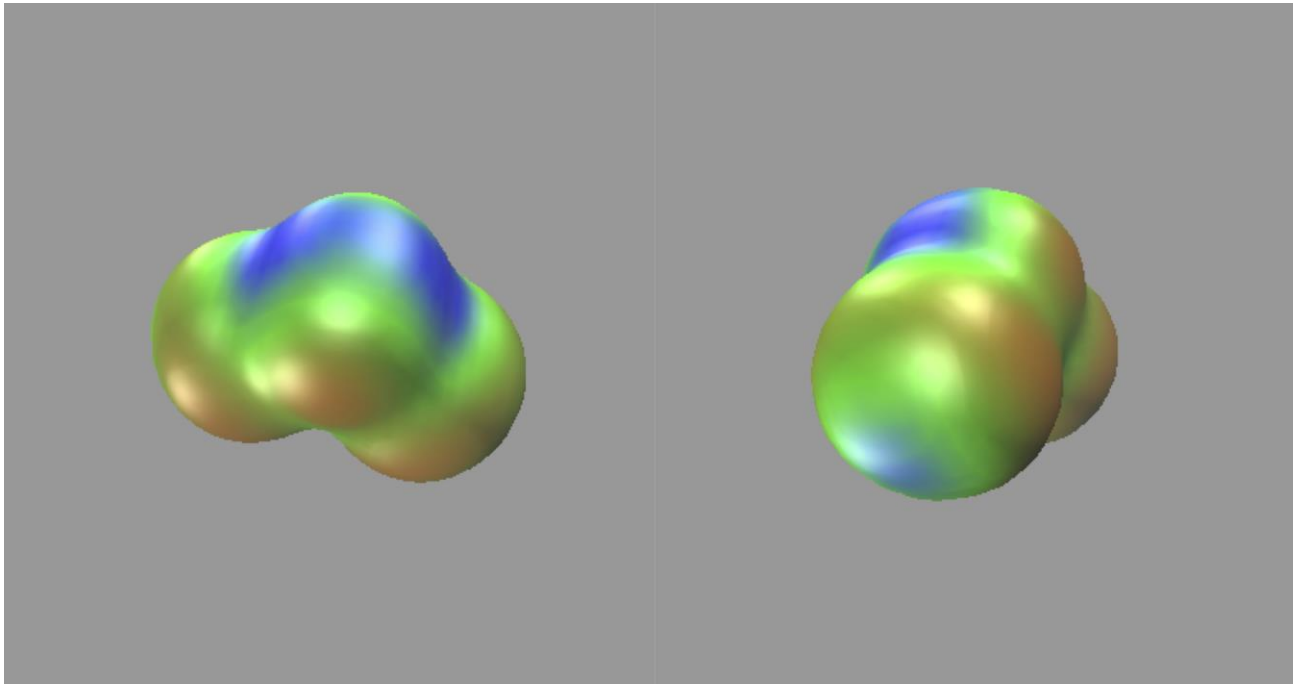
- reduced gradient; Laplacian, arbitrary-order derivatives of electron density and one-electron reduced density-matrix (1DM).
- Information-theoretic descriptors
- Electrostatic potential
- Density-based energy decomposition analysis
- Constrained Quantum Chemistry (and DFT) Analysis

Electrostatic Potential

- Positive values associated with electrophilic molecular regions
- Negative values associated with nucleophilic molecular regions

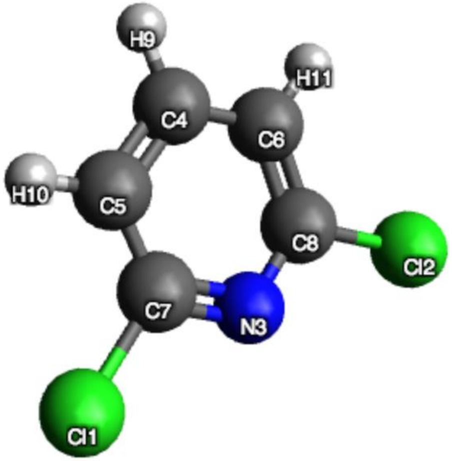
$$\Phi(\mathbf{r}) = \sum_{A=1}^{N_{\text{atoms}}} \frac{Z_A}{|\mathbf{r} - \mathbf{R}_A|} - \int \frac{\rho(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d\mathbf{r}'$$

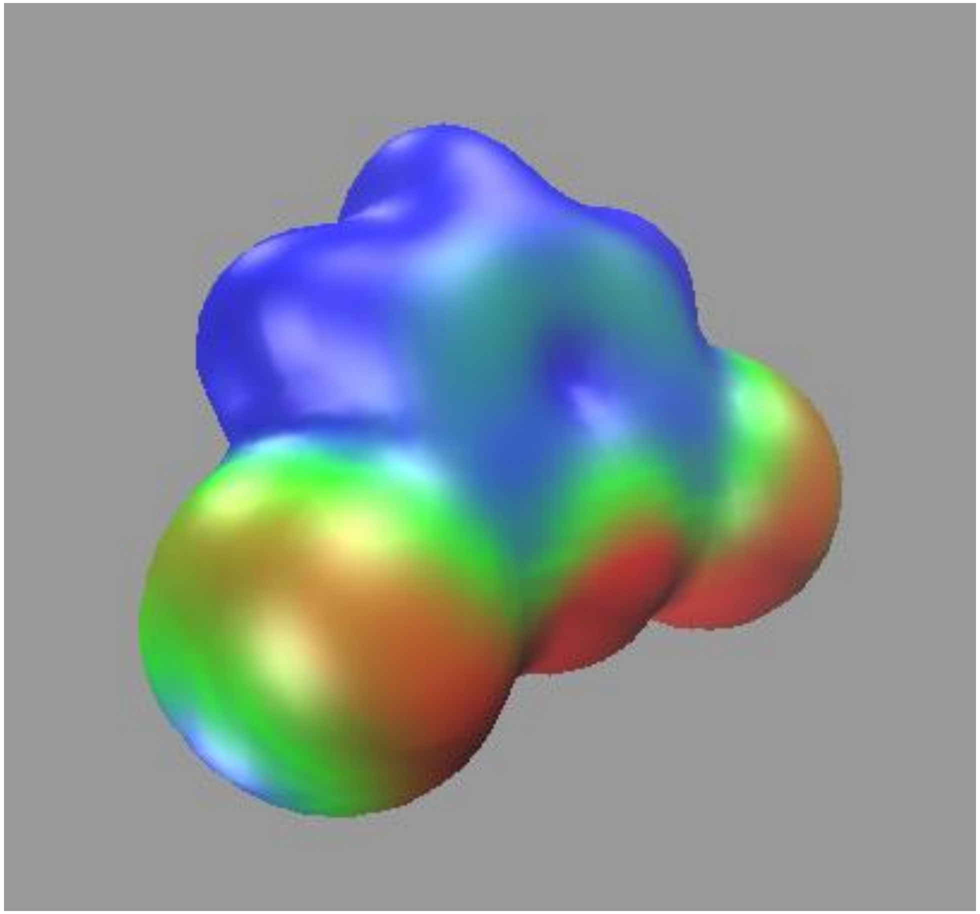
Electrostatic Potential (ESP)



```
$ chemtools esp –h

$ chemtools esp scl2.fchk --isosurface=0.002
```

Electrostatic Potential (ESP)





Python Script & Jupyter Notebooks

Included & Planned Features

(DFT-based) Descriptors for bonding and nonbonding molecular interactions

- ELF: Electron Localization Function
- LOL: Localized Orbital Locator
- DORI: Density Overlap Region Indicator
- Local Kinetic Energy
- Slater Potential, Ehrenfest Force, Stress Tensor
- NCI: Non-Covalent Interaction (reduced density gradient)

In general, these tools consist of a fundamental descriptor, ξ(**r**), which is then transformed onto a convenient interval for visualization. ChemTools provides access to the (raw) fundamental descriptors and a variety of useful mappings:

$$x_a^{(k)}(\xi) = \frac{1}{1 + a\xi^k}$$

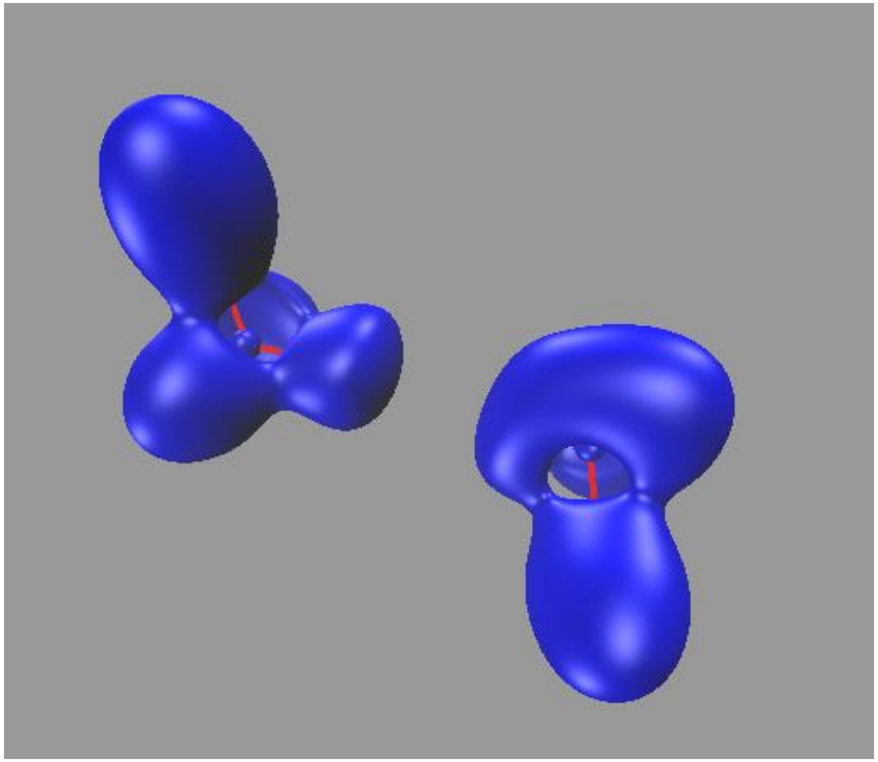
$$\tilde{x}_a^{(k)}(\xi) = 1 - x_a^{(k)}(\xi)$$

$$v_a^{(k)}(\xi) \equiv \tfrac{1}{2}\left(1 + \tanh\left(a\left(\xi^{-k} - \xi^k\right)\right)\right)$$

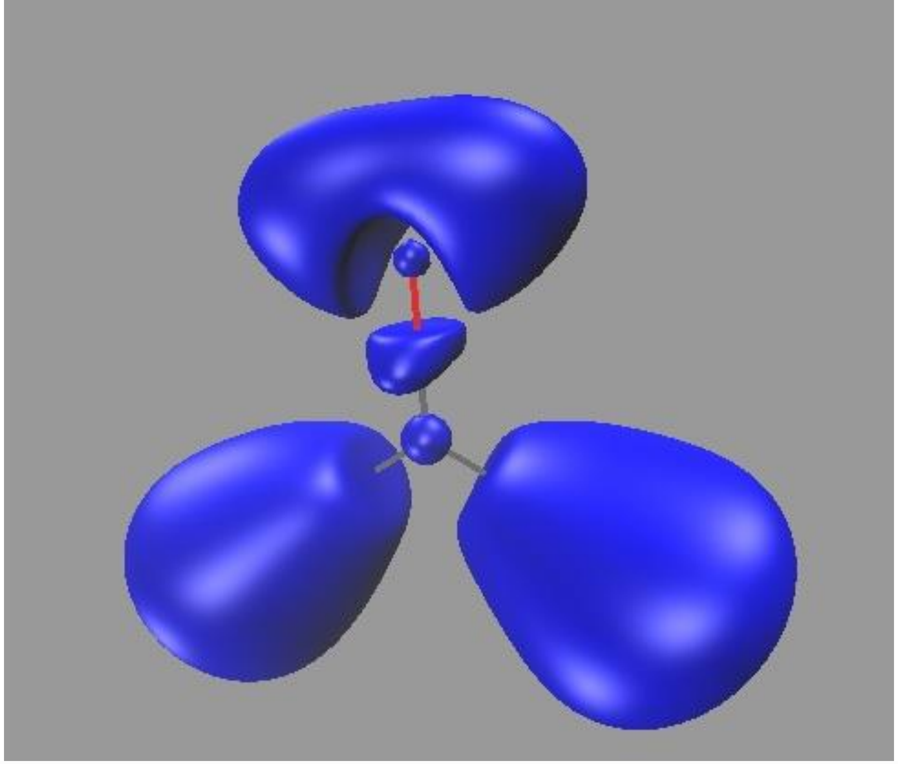
$$\tilde{v}_a^{(k)}(\xi) \equiv \tfrac{1}{2}\left(1 + \tanh\left(a\left(\xi^{k} - \xi^{-k}\right)\right)\right)$$

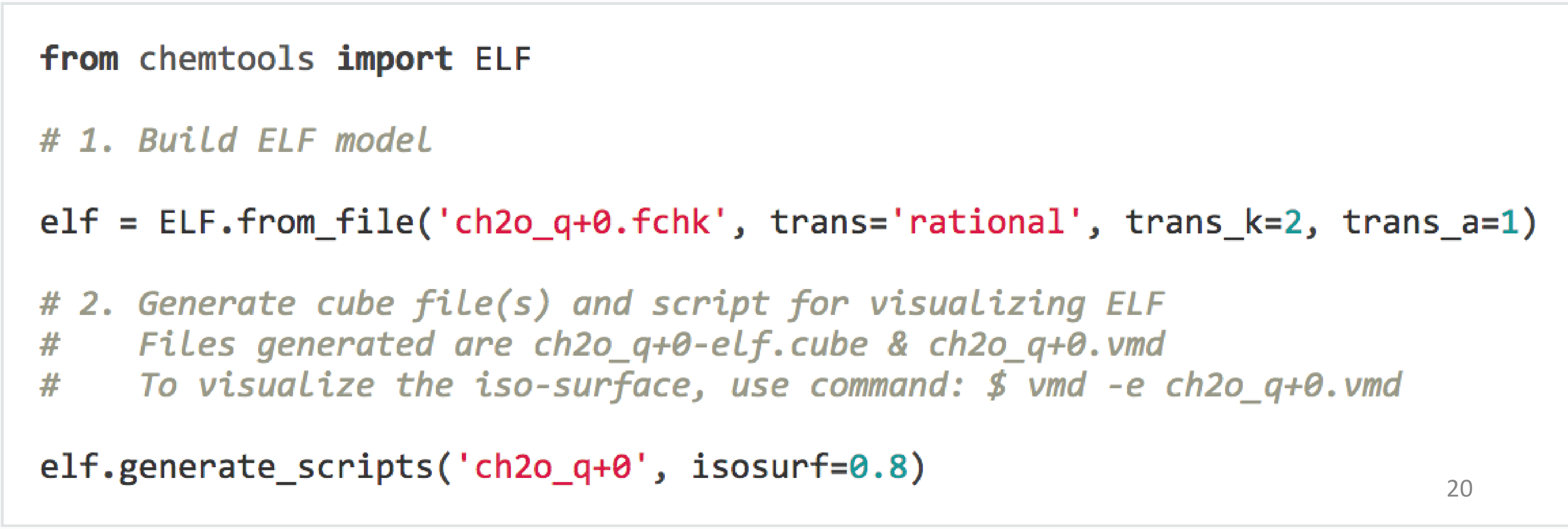
$$w_a(\xi) = \tfrac{1}{2}\left(1 + \tanh\left(a\xi\right)\right)$$

Electron Localization Function (ELF)



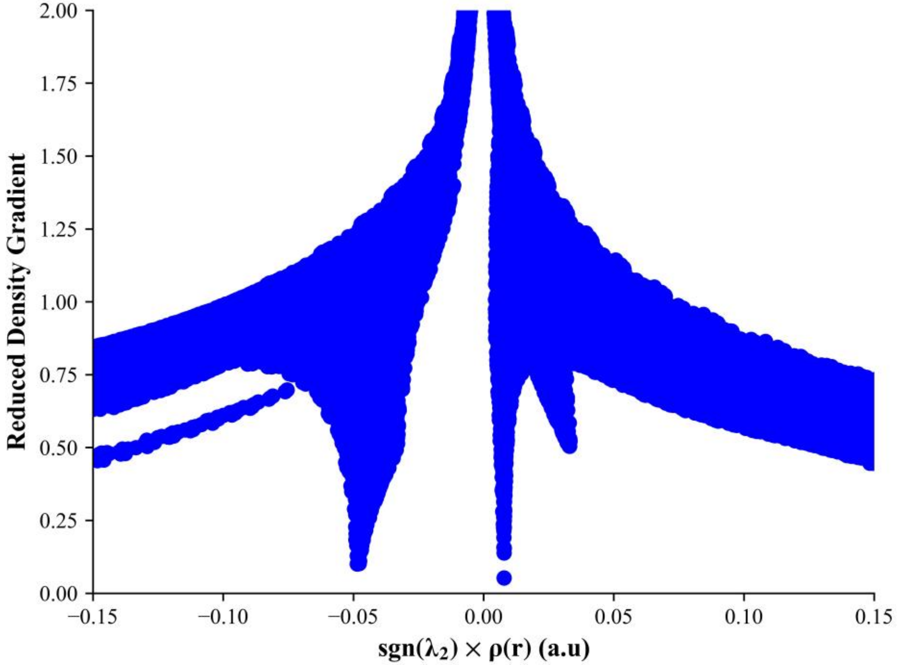
```
$ chemtools elf -h

$ chemtools elf h2o_dimer.fchk
```

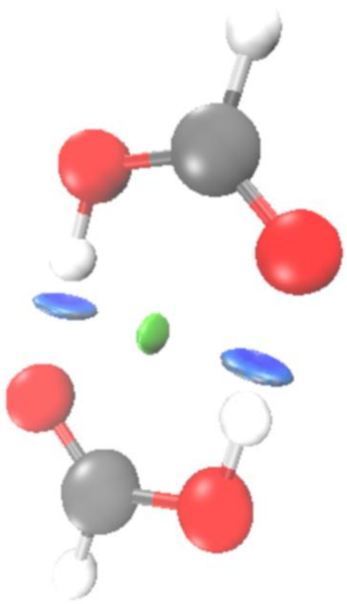




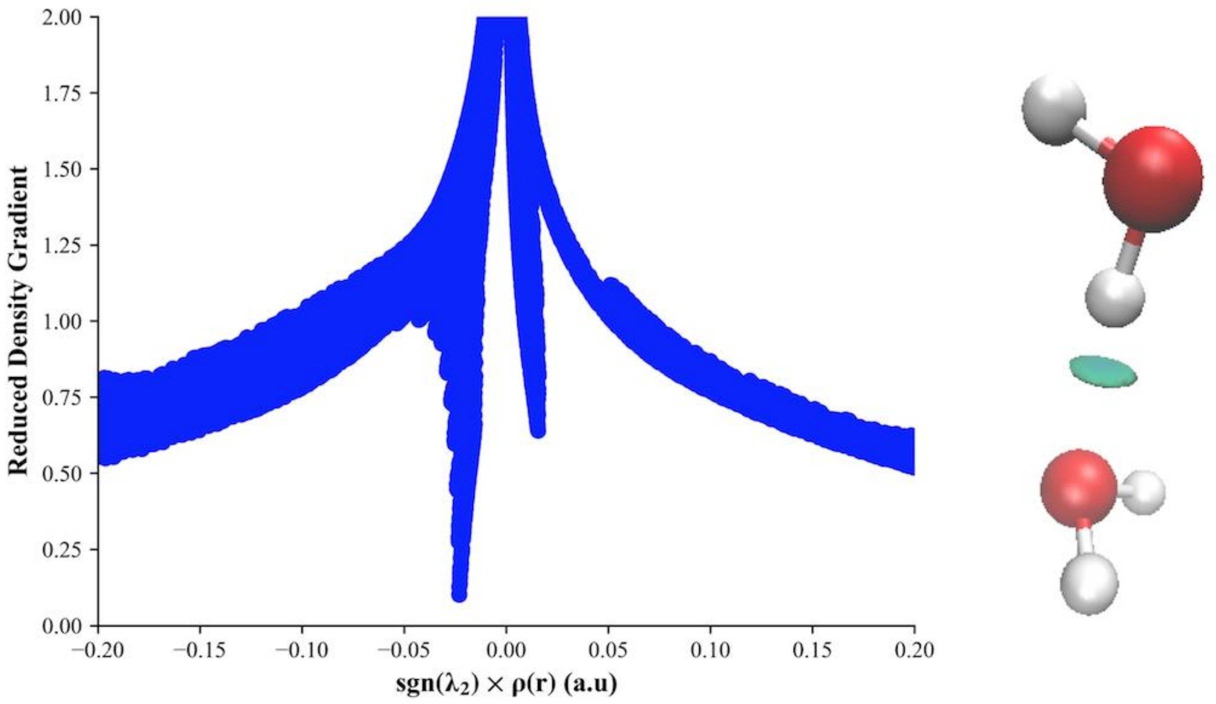
```
from chemtools import ELF

# 1. Build ELF model

elf = ELF.from_file('ch2o_q+0.fchk', trans='rational', trans_k=2, trans_a=1)

# 2. Generate cube file(s) and script for visualizing ELF
#    Files generated are ch2o_q+0-elf.cube & ch2o_q+0.vmd
#    To visualize the iso-surface, use command: $ vmd -e ch2o_q+0.vmd

elf.generate_scripts('ch2o_q+0', isosurf=0.8)
```

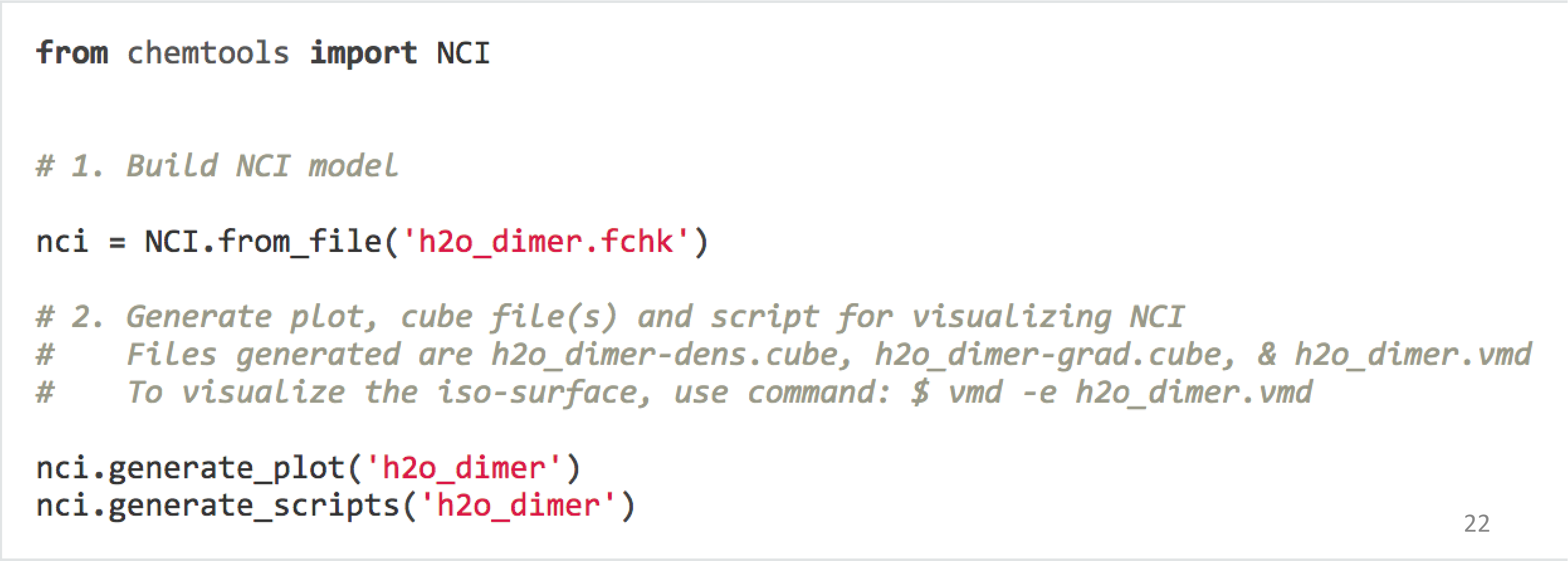
Non-Covalent Interactions (NCI)





```
$ chemtools nci -h

$ chemtools nci formic_acid_dimer.fchk
```





```
from chemtools import NCI


# 1. Build NCI model

nci = NCI.from_file('h2o_dimer.fchk')

# 2. Generate plot, cube file(s) and script for visualizing NCI
#    Files generated are h2o_dimer-dens.cube, h2o_dimer-grad.cube, & h2o_dimer.vmd
#    To visualize the iso-surface, use command: $ vmd -e h2o_dimer.vmd

nci.generate_plot('h2o_dimer')
nci.generate_scripts('h2o_dimer')
```

Included & Planned Features

Conceptual DFT Descriptors

- global, (non-)local, & condensed descriptors of arbitrary order
- popular & general (user-defined) energy models (symbolically)
- spin-reactivity indicators
- isomorphic ensemble tools (local hardness, hardness kernel, etc.)
- alchemical energies, responses, and potentials.

Key Idea: Chemical reactivity can be understood by studying how molecules respond to changes in the number of electrons and external potential.

https://chemtools.org/sci_doc_conceptual.html

Key Reactivity Indicators: Response functions. Understanding how the energy changes as the number of electrons changes requires an energy model and data about the energy for different numbers of electrons.

Included & Planned Features

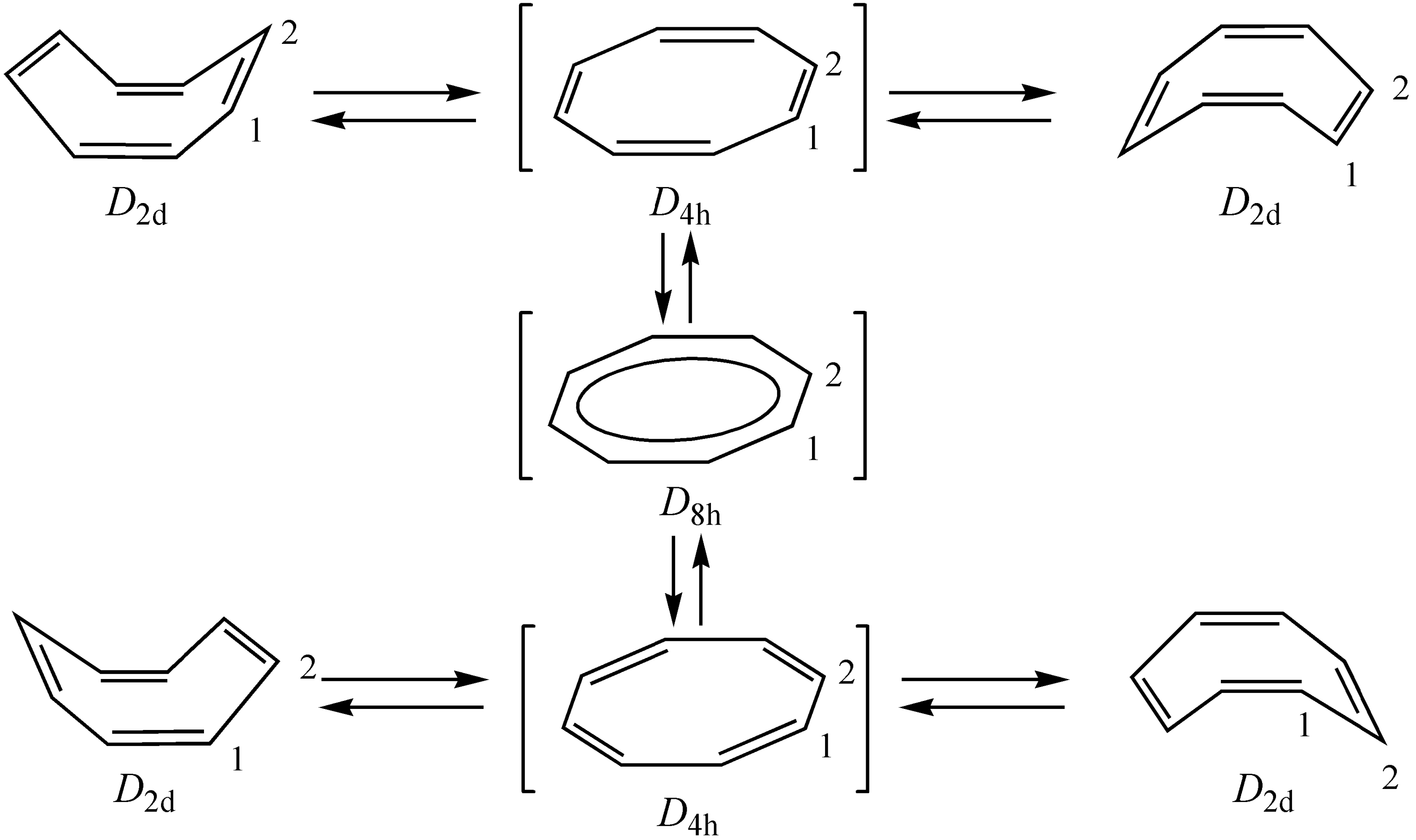
Conceptual DFT Descriptors

- global, (non-)local, & condensed descriptors of arbitrary order
- popular & general (user-defined) energy models (symbolically)
- alchemical energies, responses, and potentials.

$$\begin{tikzcd}
 & & & E[N,v(\mathbf{r})] \arrow[dl] \arrow[dr] & & & \\
 & & \left(\frac{\partial E}{\partial N}\right)_{v(\mathbf{r})} = \mu \arrow[dl] \arrow[dr] & & \left(\frac{\delta E}{\delta v(\mathbf{r})}\right)_N = \rho(\mathbf{r}) \arrow[dl] \arrow[dr] & & \\
 & \left(\frac{\partial^2 E}{\partial N^2}\right)_{v(\mathbf{r})} = \eta \arrow[dl] \arrow[dr] & & \left(\frac{\partial^2 E}{\partial N \delta v(\mathbf{r})}\right) = f(\mathbf{r}) \arrow[dl] \arrow[dr] & & \left(\frac{\delta^2 E}{\delta v(\mathbf{r}) \delta v(\mathbf{r}')}\right)_N = \chi(\mathbf{r},\mathbf{r}') \arrow[dl] \arrow[dr] & \\
\left(\frac{\partial^3 E}{\partial N^3}\right)_{v(\mathbf{r})} & & \left(\frac{\partial^3 E}{\partial N^2 \delta v(\mathbf{r})}\right) = f^{(2)}(\mathbf{r}) & & \left(\frac{\partial^3 E}{\partial N \delta v(\mathbf{r}) \delta v(\mathbf{r}')}\right) & & \left(\frac{\delta^3 E}{\delta v(\mathbf{r}) \delta v(\mathbf{r}') \delta v(\mathbf{r}'')}\right)_N
\end{tikzcd}$$

Global Conceptual DFT

(cyclooctatetraene Jahn-Teller distortion)

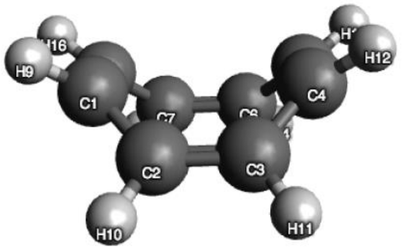


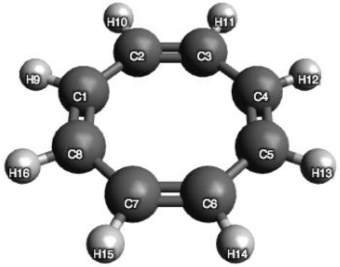
Global Conceptual DFT

(Frontier Molecular Orbital Approach)

```
$ chemtools gcdft quadratic cyclooctatetraene_planer_q+0.fchk
$ chemtools gcdft quadratic cyclooctatetraene_boat_q+0.fchk
```

```
chemical_hardness              0.155428     chemical_hardness              0.085737
chemical_potential            -0.140308     chemical_potential            -0.152984
ea                             0.062594     ea                             0.110116
electrofugality                0.281352     electrofugality                0.332341
electron_affinity              0.062594     electron_affinity              0.110116
electronegativity              0.140308     electronegativity              0.152984
electrophilicity               0.063329     electrophilicity               0.136489
eta                            0.155428     eta                            0.085737
ionization_potential           0.218022     ionization_potential           0.195852
ip                             0.218022     ip                             0.195852
mu                            -0.140308     mu                            -0.152984
n0                             56.000000    n0                             56.000000
n_max                          56.902721    n_max                          57.784350
nucleofugality                 0.000735     nucleofugality                -0.026373
softness                       6.433845     softness                       11.663633
```



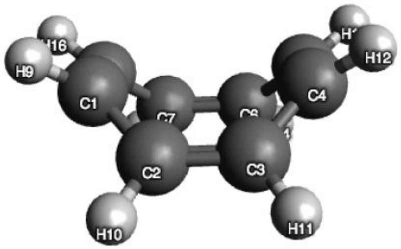


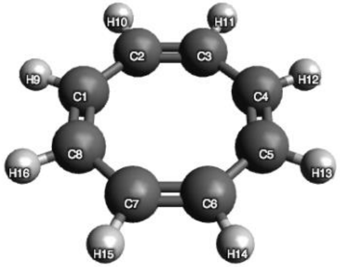
Global Conceptual DFT

(Finite Difference Approach)

```
$ chemtools gcdft quadratic cyclooctatetraene_planer_q*.fchk
$ chemtools gcdft quadratic cyclooctatetraene_boat_q*.fchk
```

```
chemical_hardness              0.294339     chemical_hardness              0.223606
chemical_potential            -0.142420     chemical_potential            -0.154832
ea                            -0.004749     ea                             0.043029
electrofugality                0.324046     electrofugality                0.320240
electron_affinity             -0.004749     electron_affinity              0.043029
electronegativity              0.142420     electronegativity              0.154832
electrophilicity               0.034456     electrophilicity               0.053605
eta                            0.294339     eta                            0.223606
ionization_potential           0.289590     ionization_potential           0.266635
ip                             0.289590     ip                             0.266635
mu                            -0.142420     mu                            -0.154832
n0                             56.000000    n0                             56.000000
n_max                          56.483864    n_max                          56.692431
nucleofugality                 0.039205     nucleofugality                 0.010576
softness                       3.397441     softness                       4.472143
```

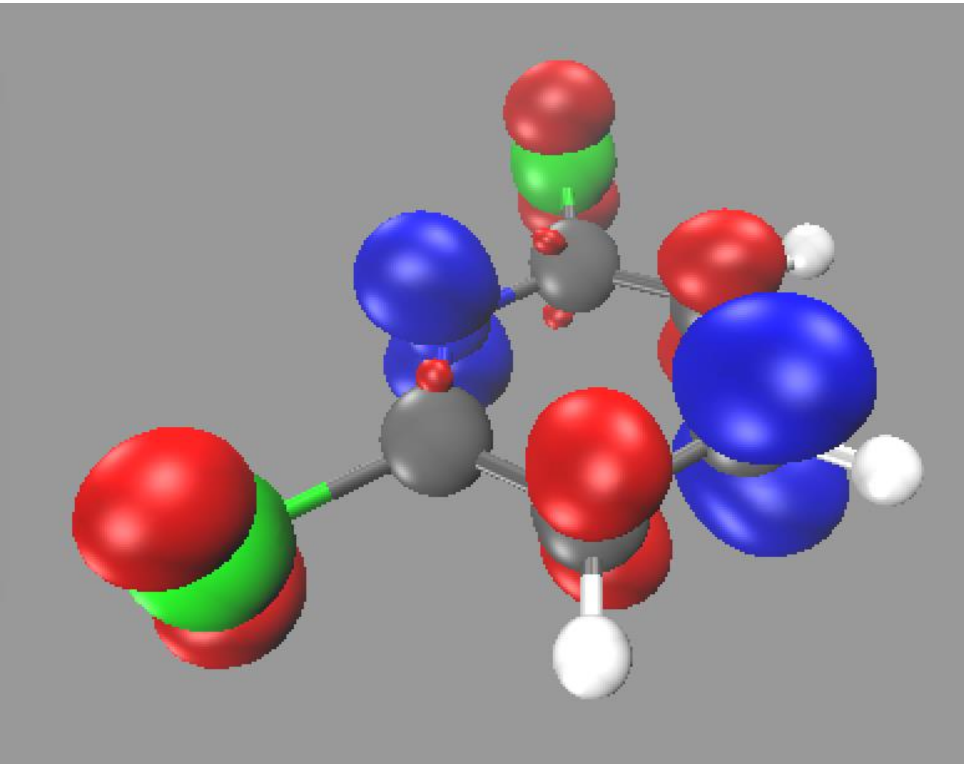




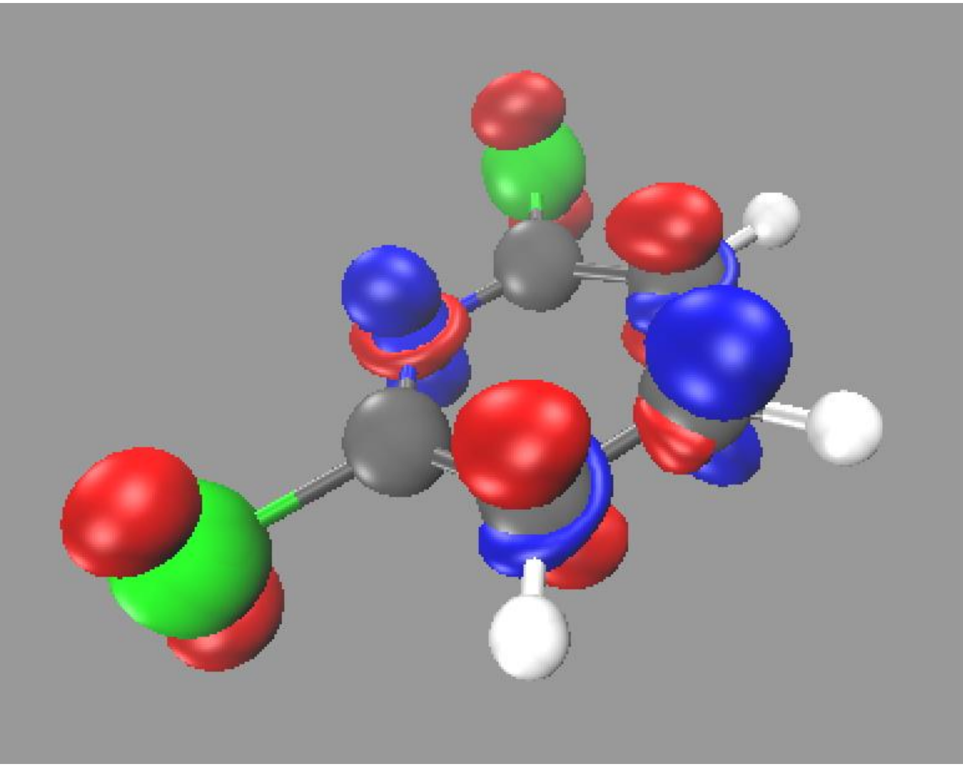
Local Conceptual DFT

(Dual Descriptor)

```
$ chemtools lcdft quadratic dual dichloropyridine_q+0.fchk
$ chemtools lcdft quadratic dual dichloropyridine_q*.fchk
```

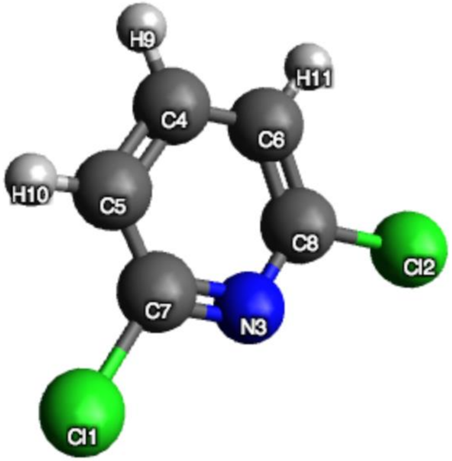


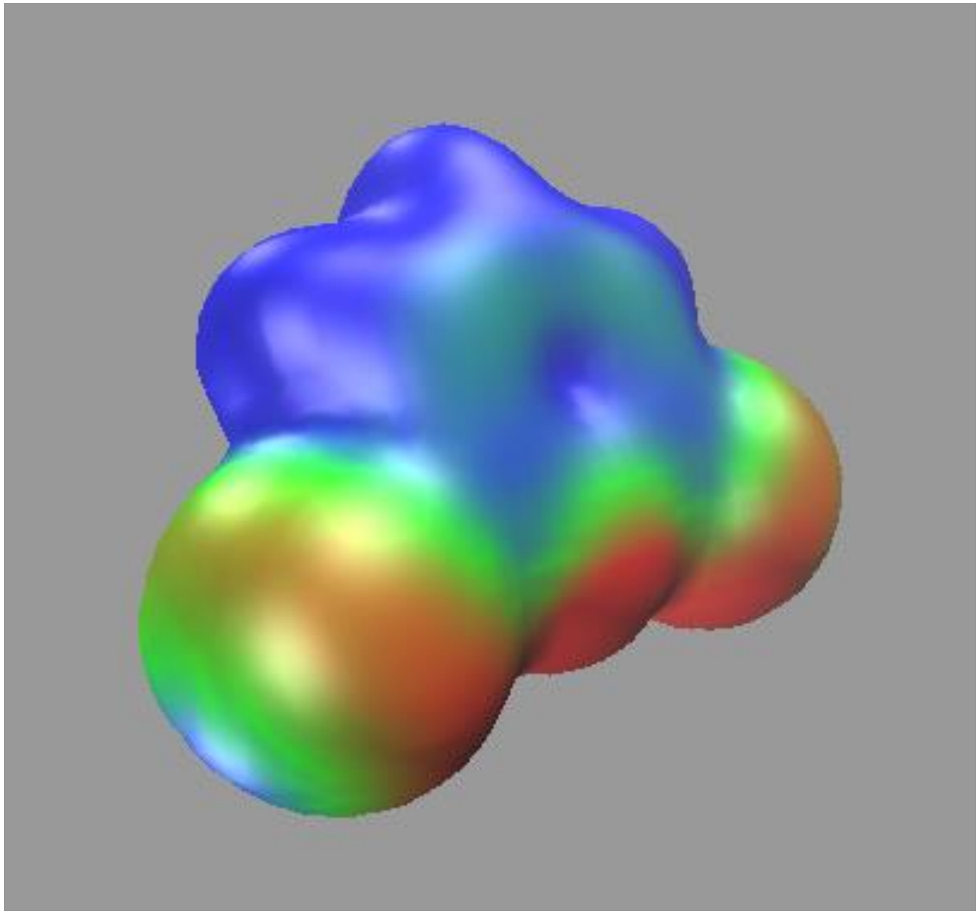
Frontier Molecular Orbital Approach



Finite Difference Approach

Electrostatic Potential (ESP)

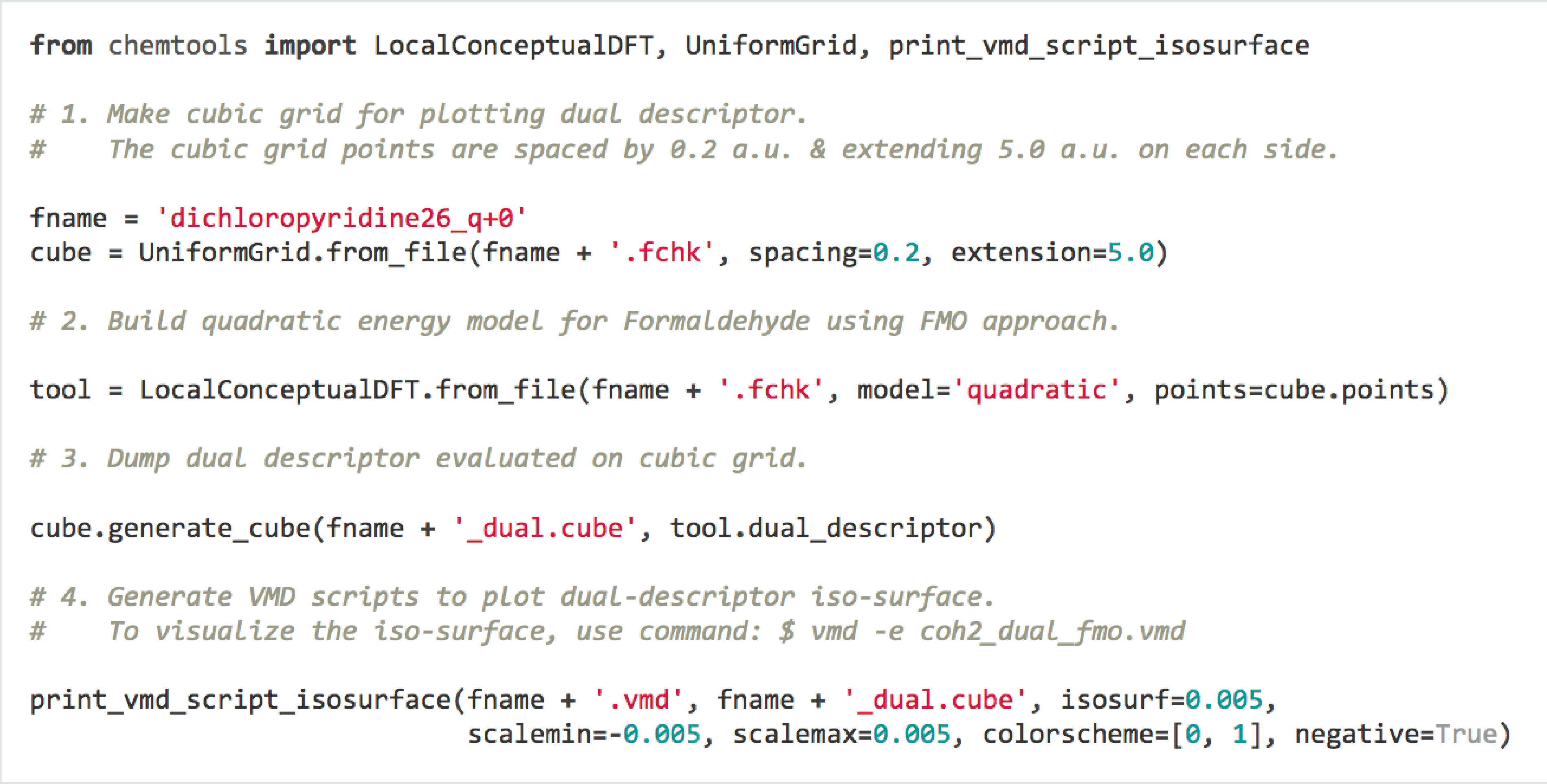




Python Script & Jupyter Notebooks

Local Conceptual DFT

(Dual Descriptor)



```
from chemtools import LocalConceptualDFT, UniformGrid, print_vmd_script_isosurface

# 1. Make cubic grid for plotting dual descriptor.
#    The cubic grid points are spaced by 0.2 a.u. & extending 5.0 a.u. on each side.

fname = 'dichloropyridine26_q+0'
cube = UniformGrid.from_file(fname + '.fchk', spacing=0.2, extension=5.0)

# 2. Build quadratic energy model for Formaldehyde using FMO approach.

tool = LocalConceptualDFT.from_file(fname + '.fchk', model='quadratic', points=cube.points)

# 3. Dump dual descriptor evaluated on cubic grid.

cube.generate_cube(fname + '_dual.cube', tool.dual_descriptor)

# 4. Generate VMD scripts to plot dual-descriptor iso-surface.
#    To visualize the iso-surface, use command: $ vmd -e coh2_dual_fmo.vmd

print_vmd_script_isosurface(fname + '.vmd', fname + '_dual.cube', isosurf=0.005,
                            scalemin=-0.005, scalemax=0.005, colorscheme=[0, 1], negative=True)
```

Included & Planned Features

Atoms-in-Molecules Partitioning

- Hirshfeld-family of partitioning (Hirshfeld, Hirshfeld-I, Minimal-Basis-Iterative-Stockholder (MBIS), Additive-Variational-Hirshfeld (AVH), etc.
- QTAIM: Quantum Theory of Atoms-in-Molecules
- Quantum Chemical Topology. (Partition with respect to arbitrary scalar/vector field.)
- Orbital-based population analysis (Mulliken/Löwdin/Ruedenberg/Knizia)

$$\rho_A^{\text{Hirshfeld}}(\mathbf{r}) = \frac{\rho_A^0(\mathbf{r})}{\sum_{B=1}^{N_{\text{atoms}}} \rho_B^0(\mathbf{r})} \rho_{\text{mol}}(\mathbf{r})$$

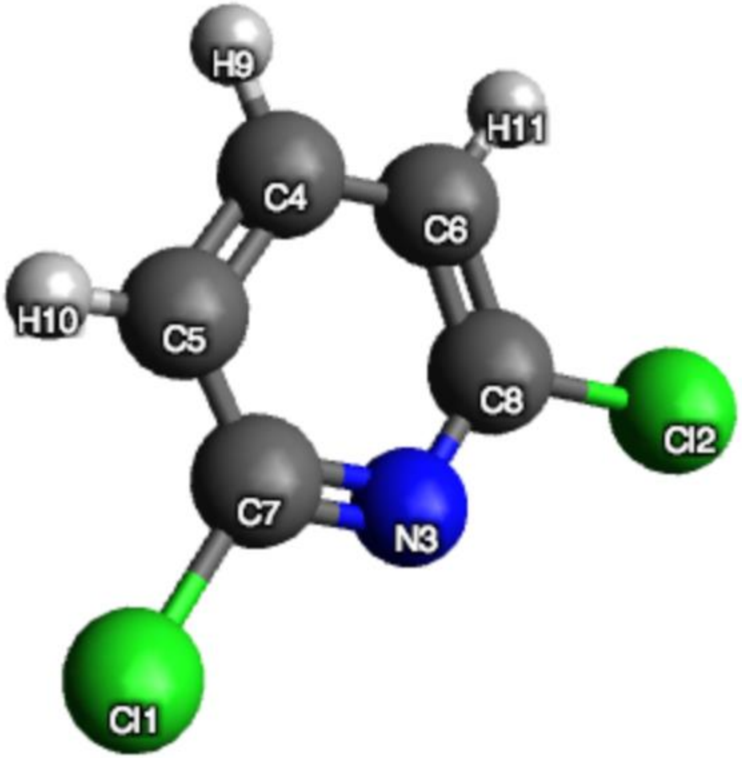
$$= \arg \min_{\rho_{\text{mol}}(\mathbf{r}) = \sum_A \rho_A(\mathbf{r})} \sum_A \int \rho_A(\mathbf{r}) \ln\left(\frac{\rho_A(\mathbf{r})}{\rho_A^0(\mathbf{r})}\right) d\mathbf{r}$$

Condensed Conceptual DFT

(Condensed Dual Descriptor)

```
$ chemtools ccdft quadratic dual dichloropyridine_q*.fchk
            --scheme mbis --approach RMF
```

```
Atomic contribution of dual_descriptor for scheme=MBIS & approach RMF:
   0     17     -0.128441
   1     17     -0.128449
   2      7      0.047067
   3      6      0.474485
   4      6     -0.413597
   5      6     -0.413582
   6      6      0.089405
   7      6      0.089391
   8      1      0.360509
   9      1      0.011791
  10      1      0.011793
```



Included & Planned Features

Density-Matrix Analysis

- Bonding indicators (SEDI, Multi-center indices)
- Effective Oxidation State (EOS)
- Intracule and Extracule
- Density Matrix, Fukui Matrix, Dual Descriptor Matrix, etc.
- Local Ionization Energy and Local Electron Affinity
- Maximum Probability Domains

Conceptual Density-Matrix Functional Theory extends Conceptual Density Functional Theory to the 1-electron reduced density matrix.

$$\frac{\partial^k \delta E}{\partial N^k \delta v(\mathbf{r})} = \left(\frac{\partial^k \rho(\mathbf{r})}{\partial N^k}\right)_{v(\mathbf{r})} \rightarrow \frac{\partial^k \delta E}{\partial N^k \delta h(\mathbf{r},\mathbf{r}')} = \left(\frac{\partial^k \gamma(\mathbf{r},\mathbf{r}')}{\partial N^k}\right)_{h(\mathbf{r},\mathbf{r}')}$$

$$\frac{\partial^k \delta E}{\partial \mu^k \delta v(\mathbf{r})} = \left(\frac{\partial^k \rho(\mathbf{r})}{\partial \mu^k}\right)_{v(\mathbf{r})} \rightarrow \frac{\partial^k \delta E}{\partial \mu^k \delta h(\mathbf{r},\mathbf{r}')} = \left(\frac{\partial^k \gamma(\mathbf{r},\mathbf{r}')}{\partial \mu^k}\right)_{h(\mathbf{r},\mathbf{r}')}$$

The effects of electron correlation of the electron density are often small, but correlation has a qualitative influence the 1DM and its derivatives.

Included & Planned Features

Density-Matrix Analysis

- Bonding indicators (SEDI, Multi-center indices)
- Effective Oxidation State (EOS)
- Intracule and Extracule
- Fukui Density Matrix, Dual Density Matrix, etc.
- Local Ionization Energy and Local Electron Affinity
- Maximum Probability Domains

The intracule and extracule show the position and extent of electron pairs

$$I_{\sigma\sigma'}\left(\mathbf{u}\right)=\iint \rho_2^{\sigma\sigma'}\left(\mathbf{r}_1,\mathbf{r}_2\right)\delta\left(\mathbf{r}_1-\mathbf{r}_2-\mathbf{u}\right)d\mathbf{r}_1 d\mathbf{r}_2$$

$$E_{\sigma\sigma'}\left(\mathbf{R}\right)=\iint \rho_2^{\sigma\sigma'}\left(\mathbf{r}_1,\mathbf{r}_2\right)\delta\left(\tfrac{1}{2}\left(\mathbf{r}_1+\mathbf{r}_2\right)-\mathbf{R}\right)d\mathbf{r}_1 d\mathbf{r}_2$$

Included & Planned Features

Density-Matrix Analysis

- Bonding indicators (SEDI, Multi-center indices)
- Effective Oxidation State (EOS)
- Intracule and Extracule
- Density Matrix, Fukui Matrix, Dual Descriptor Matrix, etc.
- Local Ionization Energy and Local Electron Affinity
- Maximum Probability Domains

Politzer's Average Local Ionization Energy is the ionization energy, per electron, at a given point in space.

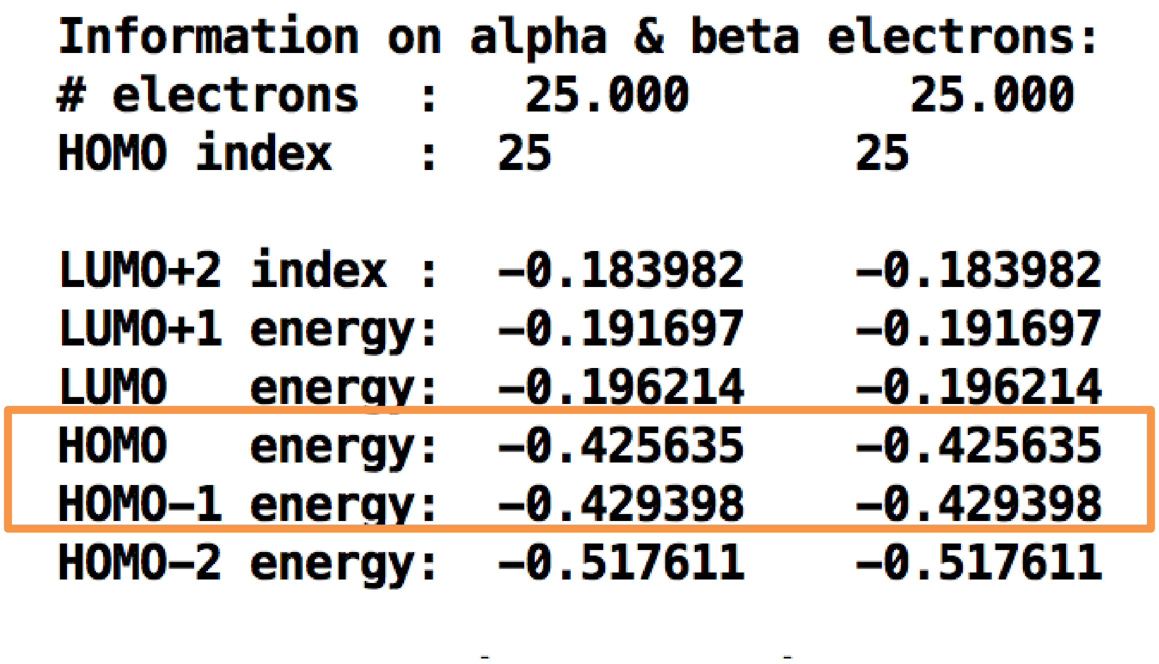
$$\text{ALIE}(\mathbf{r}) = \frac{\int\cdots\int\left(\Psi^{(N)}(\mathbf{r}_1,\mathbf{r}_2,\ldots\mathbf{r}_N)\right)^*\left(\hat{H}^{(N)}-\hat{H}^{(N-1)}\right)\Psi^{(N)}(\mathbf{r}_1,\mathbf{r}_2,\ldots\mathbf{r}_N)\,d\mathbf{r}_2\ldots d\mathbf{r}_N}{\int\cdots\int\left(\Psi^{(N)}(\mathbf{r}_1,\mathbf{r}_2,\ldots\mathbf{r}_N)\right)^*\Psi^{(N)}(\mathbf{r}_1,\mathbf{r}_2,\ldots\mathbf{r}_N)\,d\mathbf{r}_2\ldots d\mathbf{r}_N}$$

$$= \frac{\sum_{i=1}^{N_{\text{orbitals}}} n_i\varepsilon_i\left|\phi_i(\mathbf{r})\right|^2}{\sum_{i=1}^{N_{\text{orbitals}}} n_i\left|\phi_i(\mathbf{r})\right|^2}$$

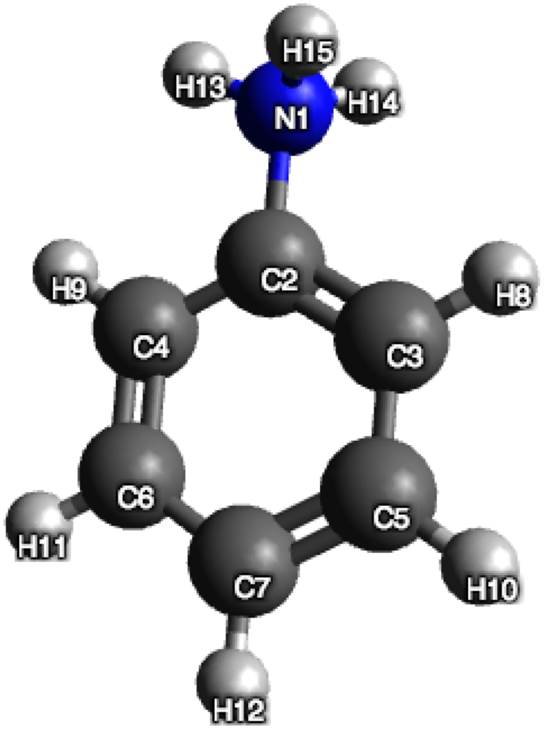
Density-Matrix Tools & Local Conceptual DFT

(Fukui Function vs. Local Ionization Potential)

```
$ chemtools mot aniline_protonated_q+1.fchk  --info
```



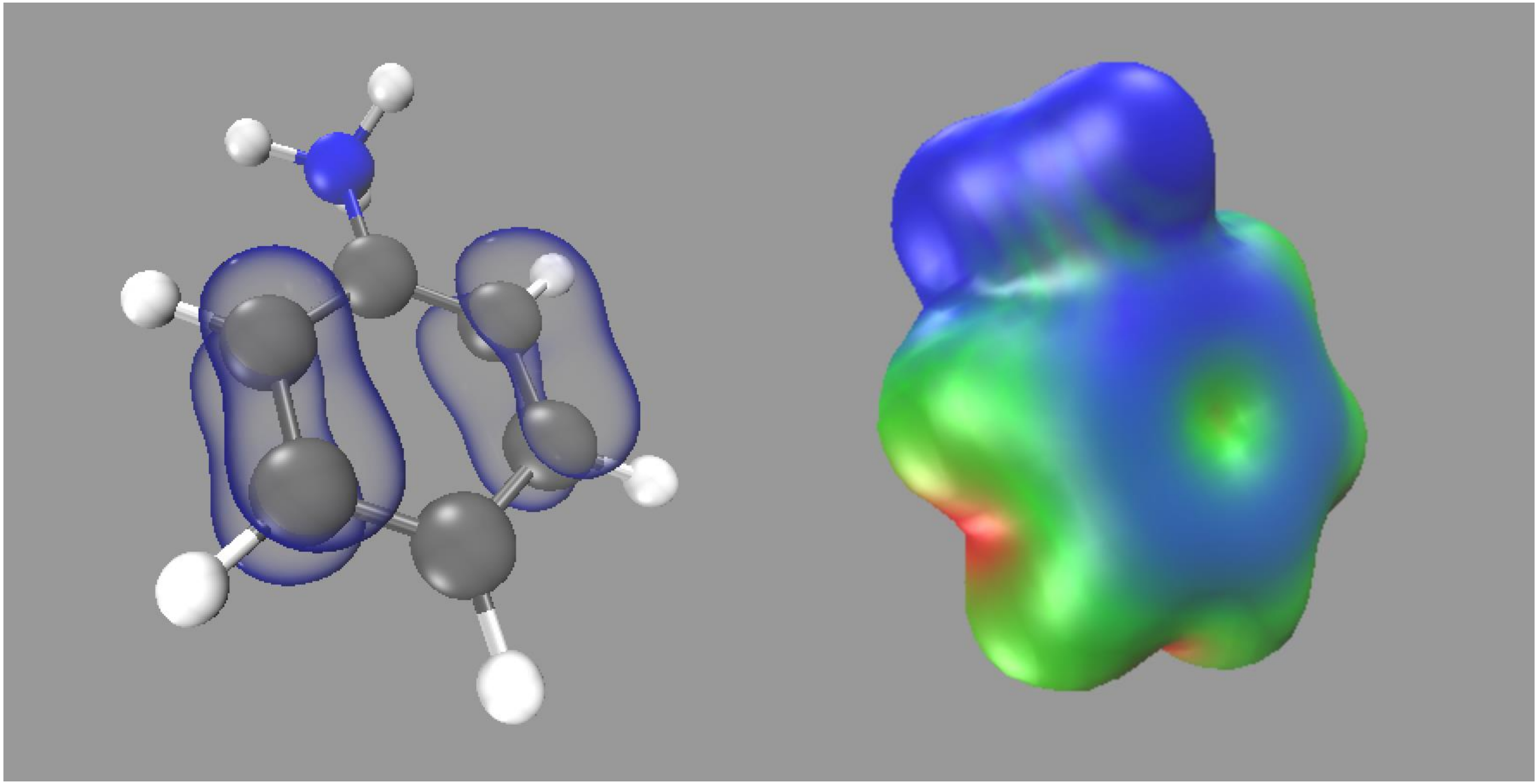
```
Information on alpha & beta electrons:
# electrons   :   25.000        25.000
HOMO index    :  25            25

LUMO+2 index :  -0.183982     -0.183982
LUMO+1 energy:  -0.191697     -0.191697
LUMO   energy:  -0.196214     -0.196214
HOMO   energy:  -0.425635     -0.425635
HOMO-1 energy:  -0.429398     -0.429398
HOMO-2 energy:  -0.517611     -0.517611
```



Density-Matrix Tools & Local Conceptual DFT

(Fukui Function vs. Local Ionization Potential)

```
$ chemtools lcdft linear ff_minus aniline_protonated_q+1.fchk
$ chemtools lip aniline_protonated_q+1.fchk
```



Included & Planned Features

Topological Analysis

- Find critical points and basins of scalars
- High-accuracy integration over basins

Take an arbitrary scalar or vector field, locate and classify critical points

$$\nabla p\left(\mathbf{r}_{\mathrm{cp}}\right) = \mathbf{0}$$

and topological basins, Ω,

$$\nabla p\left(\mathbf{r} \in \partial\Omega\right) \cdot \mathbf{n}_{\partial\Omega} = 0$$

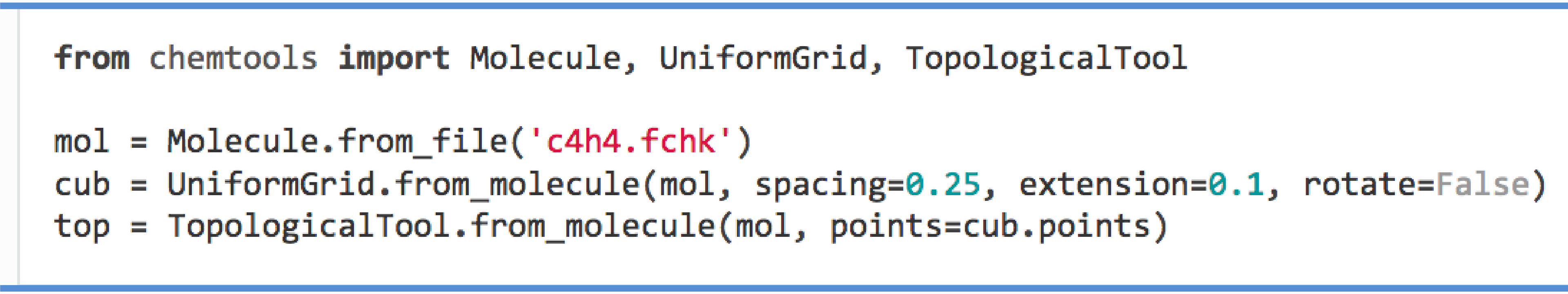
Properties at the critical points elucidate molecular interactions.

Integration of descriptors over topological basins allows one to compute energetic and other contributions from distinct chemical features.

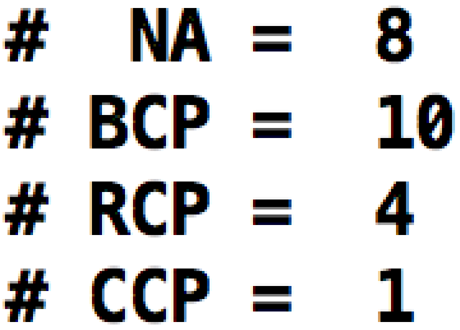
We are using new algorithms for finding critical points, for topological partitioning, and for high-accuracy integration over topological regions.

Topological Analysis

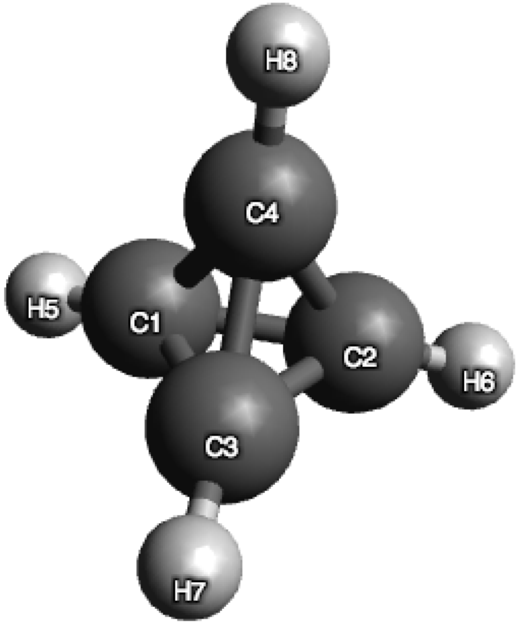
(Finding critical points of scalar fields)



```
from chemtools import Molecule, UniformGrid, TopologicalTool

mol = Molecule.from_file('c4h4.fchk')
cub = UniformGrid.from_molecule(mol, spacing=0.25, extension=0.1, rotate=False)
top = TopologicalTool.from_molecule(mol, points=cub.points)
```

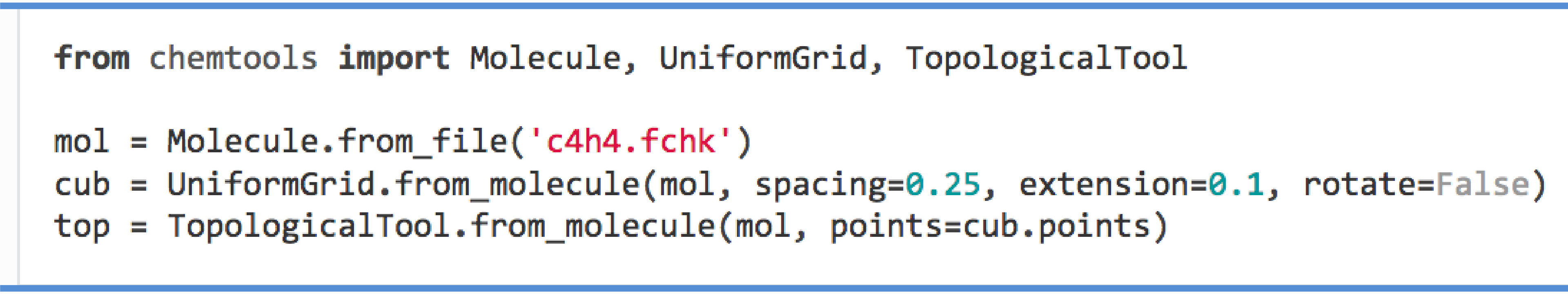


```
#  NA =  8
# BCP =  10
# RCP =  4
# CCP =  1
```

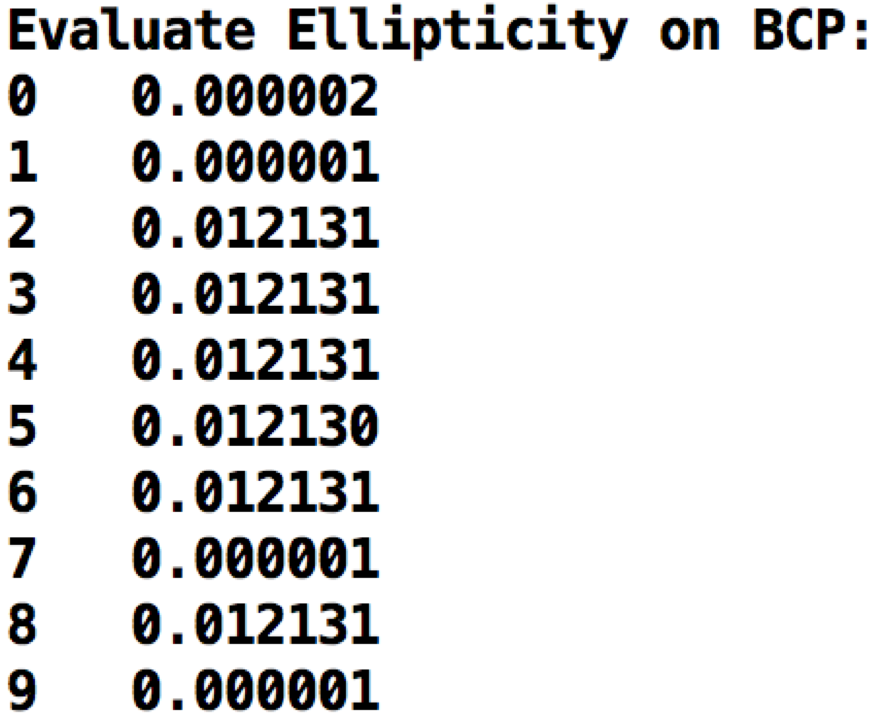


Topological Analysis

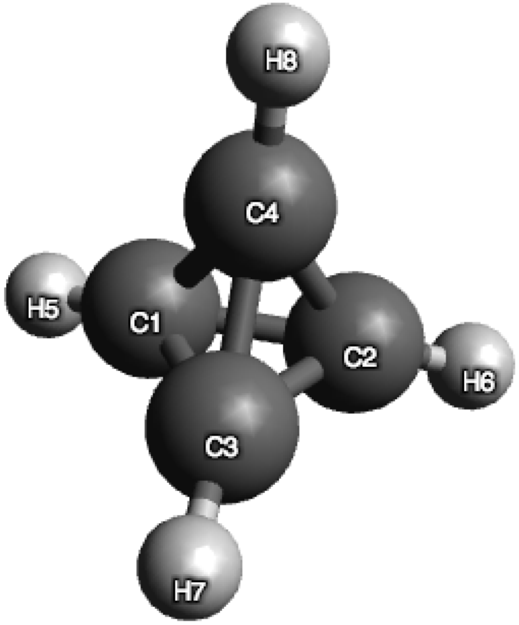
(Finding critical points of scalar fields)



```python
from chemtools import Molecule, UniformGrid, TopologicalTool

mol = Molecule.from_file('c4h4.fchk')
cub = UniformGrid.from_molecule(mol, spacing=0.25, extension=0.1, rotate=False)
top = TopologicalTool.from_molecule(mol, points=cub.points)
```



```
Evaluate Ellipticity on BCP:
0    0.000002
1    0.000001
2    0.012131
3    0.012131
4    0.012131
5    0.012130
6    0.012131
7    0.000001
8    0.012131
9    0.000001
```



Xiaotian (Derrick) Yang

Alireza Tehrani

Stijn Fias

Farnaz Heidar-Zadeh

Michael Richer

Taewon (David) Kim

Esteban
Vohringer-Martinez