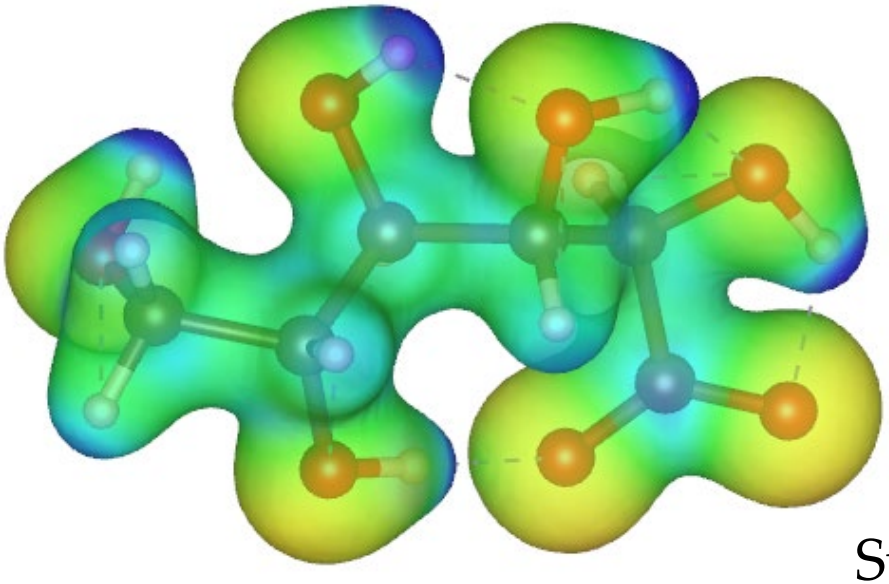
Computational Chemistry:

Hands-on Electronic Structure Calculations



Youngsam Kim

Suwon, April 26 - 28, 2023

대한화학회 제131회 학술발표회

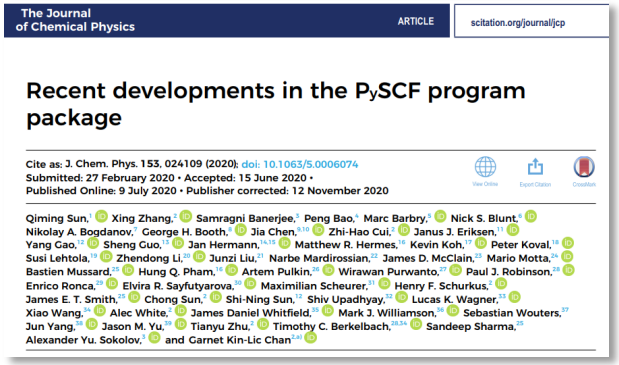
This file is uploaded to tccl.yonsei.ac.kr
and the copyright belongs to TCCL

Introduction: PySCF


Software Focus

PySCF: the Python-based simulations of chemistry framework

Qiming Sun,[1*] Timothy C. Berkelbach,[2] Nick S. Blunt,[3,4] George H. Booth,[5] Sheng Guo,[1,6] Zhendong Li,[1] Junzi Liu,[7] James D. McClain,[1,6] Elvira R. Sayfutyarova,[1,6] Sandeep Sharma,[8] Sebastian Wouters[9] and Garnet Kin-Lic Chan[1*]


The Journal of Chemical Physics ARTICLE scitation.org/journal/jcp

Recent developments in the PySCF program package

Cite as: J. Chem. Phys. 153, 024109 (2020); doi: 10.1063/5.0006074
Submitted: 27 February 2020 • Accepted: 15 June 2020 •
Published Online: 9 July 2020 • Publisher corrected: 12 November 2020

Qiming Sun, Xing Zhang, Samragni Banerjee, Peng Bao, Marc Barbry, Nick S. Blunt, Nikolay A. Bogdanov, George H. Booth, Jia Chen, Zhi-Hao Cui, Janus J. Eriksen, Yang Gao, Sheng Guo, Jan Hermann, Matthew R. Hermes, Kevin Koh, Peter Koval, Susi Lehtola, Zhendong Li, Junzi Liu, Narbe Mardirossian, James D. McClain, Mario Motta, Bastien Mussard, Hung Q. Pham, Artem Pulkin, Wirawan Purwanto, Paul J. Robinson, Enrico Ronca, Elvira R. Sayfutyarova, Maximilian Scheurer, Henry F. Schurkus, James E. T. Smith, Chong Sun, Shi-Ning Sun, Shiv Upadhyay, Lucas K. Wagner, Xiao Wang, Alec White, James Daniel Whitfield, Mark J. Williamson, Sebastian Wouters, Jun Yang, Jason M. Yu, Tianyu Zhu, Timothy C. Berkelbach, Sandeep Sharma, Alexander Yu. Sokolov, and Garnet Kin-Lic Chan

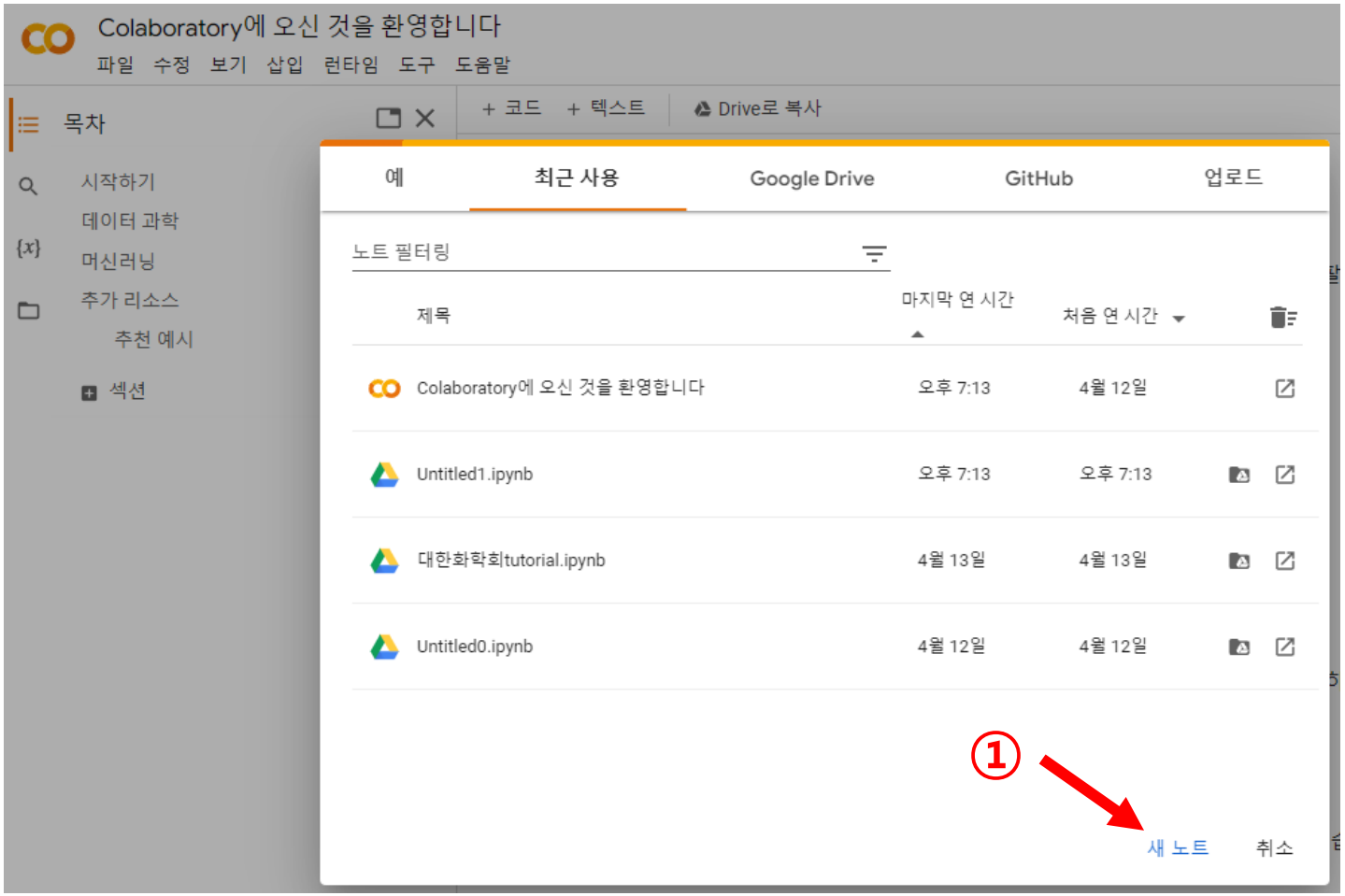
Python-based **S**imulations of **C**hemistry **F**ramework

- Open-source
- Python-based (intuitive)
- Gaussian type orbital(GTO) for cluster systems
- GTO + planewave for periodic systems

Q. Sun, et al., WIREs Comput. Mol. Sci. 8, e1340 (2018)
Q. Sun, et al., J. Chem. Phys. 153, 024109 (2020)

Introduction: Settings

① Google Colab & 새 노트 click



Introduction: Settings

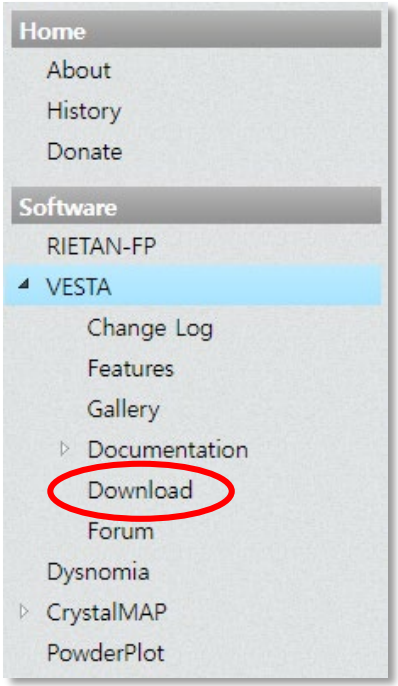
② Define/Modify note title

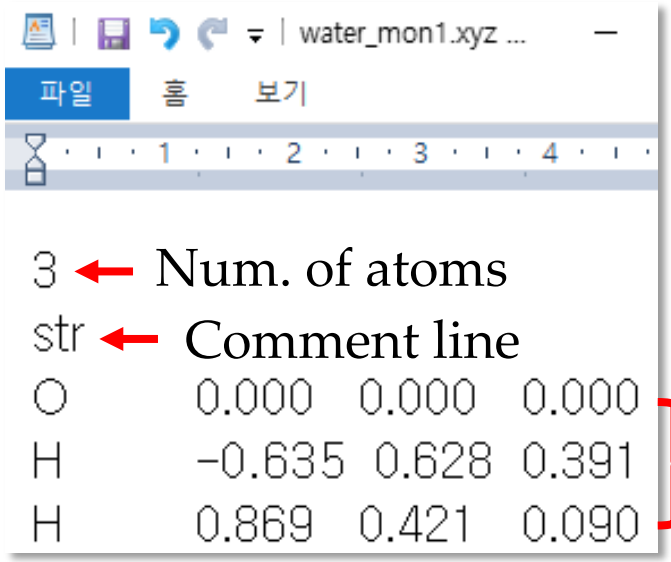
③ PySCF installation → pip install pyscf

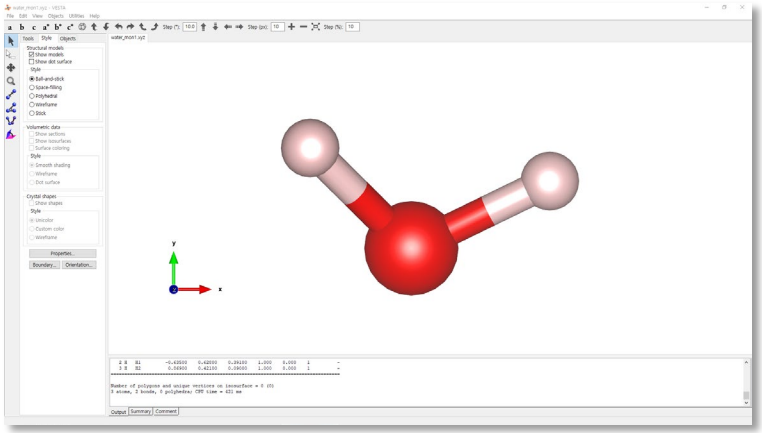
④ Add 코드

Introduction: Settings

⑤ VESTA (**V**isulaization for **E**lectronic and **ST**ructural **A**nalysis) (optional)







Element X Y Z

워드패드(or 메모장) → save *.xyz

K. Momma, et al., J. Appl. Cryst. 44, 1272-1276 (2011)

Introduction: Settings

Part I. Basic Density Functional Theory Calculation

Input & Output for H_2O monomer

Exercise) H_2O dimer interaction energy

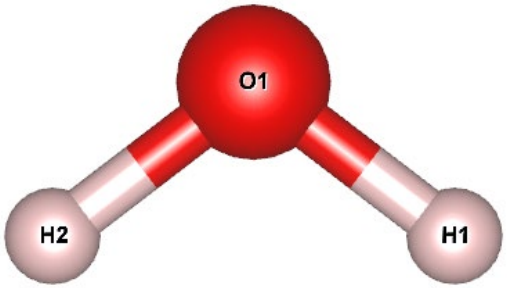
Part II. Geometry Optimization

Input & Output for LiH *database

Exercise) CH_3OH optimization

Part III. Dispersion Correction

Part I. Basic DFT Calculation



<Input code>

```
from pyscf import gto, dft

mol = gto.M(
    atom='''
    O 0  0  0
    H 0  0.77  0.58
    H 0  -0.77  0.58''',
    spin=0,
    charge=0,
    basis= 'ccpvdz',
    verbose=4)
```

①

```
mydft = dft.UKS(mol)
mydft.xc = 'b3lyp'
mydft.kernel()

mydft.analyze()
```

②

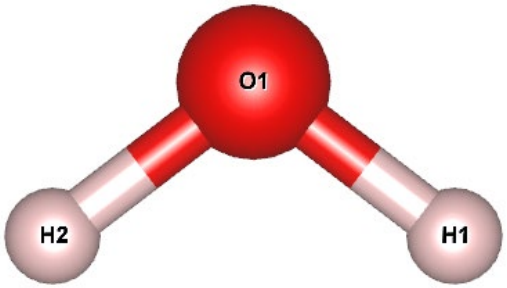
!! Take caution !!

bracket (), spacing,

comma, enter ...

Run → Click or Ctrl+Enter

Part I. Basic DFT Calculation



<Input code>

```
from pyscf import gto, dft
```

Import package

```
mol = gto.M(
    atom='''
    O 0  0  0
    H 0  0.77  0.58
    H 0  -0.77  0.58''',
    spin=0,
    charge=0,
    basis= 'ccpvdz',
    verbose=4)
```

Define 'mol' from 'gto'

→ atom : nuc. type, coordinates (xyz, Å)

→ spin, charge: spin state(2S) and charge

→ basis : basis sets ⇒ Prof. Sherrill's note

* http://vergil.chemistry.gatech.edu/notes/

→ Verbose : print level(4)

```
mydft = dft.UKS(mol)
mydft.xc = 'b3lyp'
mydft.kernel()

mydft.analyze()
```

→ (UKS)RKS : (Un-)Restricted Kohn Sham

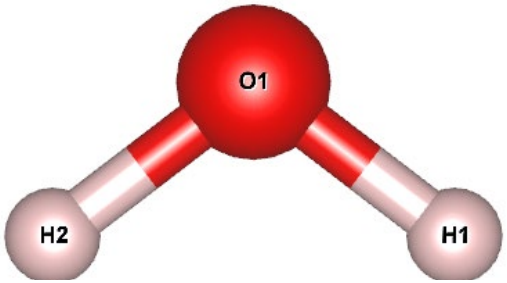
w/o α, β distinction. orbital occupation = 2 or 0

→ kernel() : perform calculation

→ analyze() : population analyze

Part I. Basic DFT Calculation

<Output>




```
System: uname_result(system='Linux', node='4149422508eb', release='5.10.147+', version='#1 SMP Sat Dec 10 16:00:40 UTC 2022', machine='x86_64')  Threads 2
Python 3.9.16 (main, Dec  7 2022, 01:11:51)
[GCC 9.4.0]
numpy 1.22.4  scipy 1.10.1
Date: Tue Apr 25 03:55:53 2023
PySCF version 2.2.1
PySCF path  /usr/local/lib/python3.9/dist-packages/pyscf

[CONFIG] conf_file None
[INPUT] verbose = 4
[INPUT] num. atoms = 3
[INPUT] num. electrons = 10
[INPUT] charge = 0
[INPUT] spin (= nelec alpha-beta = 2S) = 0
[INPUT] symmetry False subgroup None
[INPUT] Mole.unit = angstrom
[INPUT] Symbol           X                Y                Z      unit         X                Y                Z      unit  Magmom
[INPUT]  1 O      0.000000000000   0.000000000000   0.000000000000 AA    0.000000000000   0.000000000000   0.000000000000 Bohr   0.0
[INPUT]  2 H      0.000000000000   0.770000000000   0.580000000000 AA    0.000000000000   1.455089115915   1.096041152248 Bohr   0.0
[INPUT]  3 H      0.000000000000  -0.770000000000   0.580000000000 AA    0.000000000000  -1.455089115915   1.096041152248 Bohr   0.0

nuclear repulsion = 9.12662691069514
number of shells = 11
number of NR pGTOs = 40
number of NR cGTOs = 24
basis = ccpvdz
ecp = {}
CPU time:         9.60

number electrons alpha = 5  beta = 5
XC library pyscf.dft.libxc version 6.1.0
    S. Lehtola, C. Steigemann, M. J.T. Oliveira, and M. A.L. Marques.,  SoftwareX 7, 1-5 (2018)
XC functionals = b3lyp
    P. A. M. Dirac.,  Math. Proc. Cambridge Philos. Soc. 26, 376 (1930)
    F. Bloch.,  Z. Phys. 57, 545 (1929)
    A. D. Becke.,  Phys. Rev. A 38, 3098 (1988)
    C. Lee, W. Yang, and R. G. Parr.,  Phys. Rev. B 37, 785 (1988)
    B. Miehlich, A. Savin, H. Stoll, and H. Preuss.,  Chem. Phys. Lett. 157, 200 (1989)
    S. H. Vosko, L. Wilk, and M. Nusair.,  Can. J. Phys. 58, 1200 (1980)
```

- Check if the input is **well defined as the user thinks/wants!**

Part I. Basic DFT Calculation

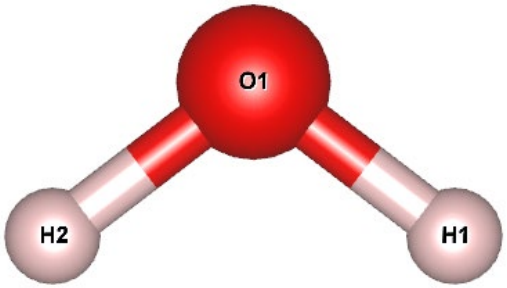
<Input code> in Gaussian 16

```
#p
#bv5lyp/cc-pvdz iop(3/76=1000002000) iop(3/77=0720008000) iop(3/78=0810010000)

title

0 1
O   0.   0.   0.
H   0.   0.77   0.58
H   0.   -0.77   0.58
```

<Input code> in ORCA

```
! B3LYP cc-pvdz
* xyz 0 1
 O   0.   0.   0.
 H   0.   0.77   0.58
 H   0.   -0.77   0.58
*
```

Part I. Basic DFT Calculation



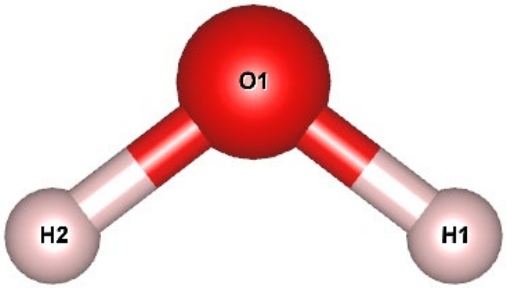
<Output>

```
init E= -76.3176487258378
  HOMO = -0.424214880461059  LUMO = 0.00576850782758609
cycle= 1 E= -76.2907860084321  delta_E= 0.0269  |g|= 0.711  |ddm|= 1.37
  HOMO = -0.115580962229854  LUMO = 0.0866957495862257
cycle= 2 E= -76.2059987829547  delta_E= 0.0848  |g|= 0.983  |ddm|= 0.973
  HOMO = -0.28894456934566  LUMO = 0.0500835157204298
cycle= 3 E= -76.3830990955477  delta_E= -0.177  |g|= 0.0234  |ddm|= 0.588
  HOMO = -0.283089328138573  LUMO = 0.054203061741246
cycle= 4 E= -76.3831985069832  delta_E= -9.94e-05  |g|= 0.00375  |ddm|= 0.0136
  HOMO = -0.283413378722867  LUMO = 0.0538996564196925
cycle= 5 E= -76.3832007801309  delta_E= -2.27e-06  |g|= 0.000462  |ddm|= 0.00203
  HOMO = -0.2835551130816 61  LUMO = 0.0538724659684704
cycle= 6 E= -76.3832008204877  delta_E= -4.04e-08  |g|= 7.88e-06  |ddm|= 0.000284
  HOMO = -0.283555920718238  LUMO = 0.0538710838505195
cycle= 7 E= -76.3832008205172  delta_E= -2.95e-11  |g|= 1.18e-06  |ddm|= 1.38e-05
  HOMO = -0.283555604005165  LUMO = 0.0538712791419524
Extra cycle  E= -76.3832008205176  delta_E= -4.12e-13  |g|= 7.73e-07  |ddm|= 1.57e-06
converged SCF energy = -76.3832008205176
```

HOMO, LUMO → Gap

Total energy [*Eh*]

Part I. Basic DFT Calculation



<Output>

```
**** SCF Summaries ****
Total Energy =                              -76.383200820517644
Nuclear Repulsion Energy =                    9.126626910695141
One-electron Energy =                      -123.103556146575698
Two-electron Coulomb Energy =                46.923017812384003
DFT Exchange-Correlation Energy =            -9.329289397021089
**** MO energy ****
MO #1   energy= -19.1186491430855  occ= 2
MO #2   energy= -0.987661195417317 occ= 2
MO #3   energy= -0.510664437864799 occ= 2
MO #4   energy= -0.357633758735553 occ= 2
MO #5   energy= -0.283555604005165 occ= 2
MO #6   energy= 0.0538712791419524 occ= 0
MO #7   energy= 0.129377406916558  occ= 0
MO #8   energy= 0.565111003576567  occ= 0
MO #9   energy= 0.606639719197764  occ= 0
MO #10  energy= 0.903258789871577  occ= 0
MO #11  energy= 0.922489891876981  occ= 0
MO #12  energy= 0.997298355595242  occ= 0
MO #13  energy= 1.22165464571497   occ= 0
MO #14  energy= 1.23085906956994   occ= 0
MO #15  energy= 1.43785119825297   occ= 0
MO #16  energy= 1.60095726644132   occ= 0
MO #17  energy= 1.6599205573074    occ= 0
MO #18  energy= 2.11013769008181   occ= 0
MO #19  energy= 2.1413010108255    occ= 0
MO #20  energy= 2.91220076954952   occ= 0
MO #21  energy= 2.96986766410649   occ= 0
MO #22  energy= 3.12174633634056   occ= 0
MO #23  energy= 3.47733786606679   occ= 0
MO #24  energy= 3.72564591069997   occ= 0
```

RKS

UKS

$$* E_{tot} = E_{NN} + T + E_{Ne} + J + E_{XC}$$

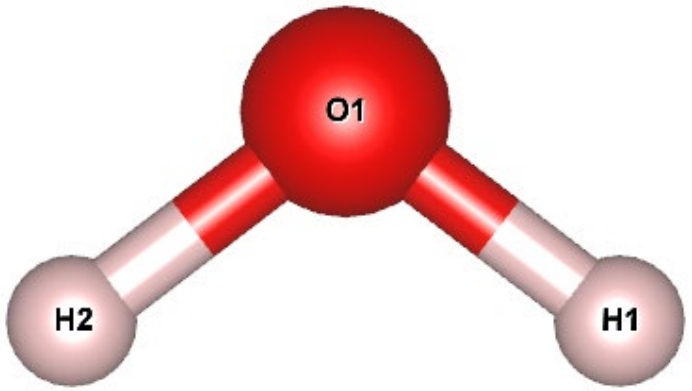
```
**** MO energy ****
                                alpha | beta                 alpha | beta
MO #1   energy= -19.11864969637    | -19.1186496461307  occ= 1 | 1
MO #2   energy= -0.987661462457649 | -0.987661372193143 occ= 1 | 1
MO #3   energy= -0.510664614938504 | -0.510664605726537 occ= 1 | 1
MO #4   energy= -0.357633943450735 | -0.357633831652488 occ= 1 | 1
MO #5   energy= -0.283555801071953 | -0.283555756823679 occ= 1 | 1
MO #6   energy= 0.0538711765905053 | 0.0538711791352494 occ= 0 | 0
MO #7   energy= 0.129377341036489  | 0.129377321194523  occ= 0 | 0
MO #8   energy= 0.565110929334846  | 0.565110851128378  occ= 0 | 0
MO #9   energy= 0.606639581274333  | 0.606639541955059  occ= 0 | 0
MO #10  energy= 0.903258653429998  | 0.90325869870356   occ= 0 | 0
MO #11  energy= 0.9224897322983    | 0.92248975994865   occ= 0 | 0
MO #12  energy= 0.997298267744648  | 0.997298291328731  occ= 0 | 0
MO #13  energy= 1.22165458103591   | 1.22165453335582   occ= 0 | 0
MO #14  energy= 1.23085895447495   | 1.2308588830455    occ= 0 | 0
MO #15  energy= 1.43785111432482   | 1.43785104947603   occ= 0 | 0
MO #16  energy= 1.60095713547502   | 1.60095715504632   occ= 0 | 0
MO #17  energy= 1.65992047448728   | 1.65992038796426   occ= 0 | 0
MO #18  energy= 2.11013752941266   | 2.11013749434837   occ= 0 | 0
MO #19  energy= 2.14130082015479   | 2.14130080322529   occ= 0 | 0
MO #20  energy= 2.91220059386519   | 2.91220065570533   occ= 0 | 0
MO #21  energy= 2.96986750213131   | 2.96986751186434   occ= 0 | 0
MO #22  energy= 3.1217461533732    | 3.12174623431986   occ= 0 | 0
MO #23  energy= 3.477337702458     | 3.47733770897637   occ= 0 | 0
MO #24  energy= 3.7256457306752    | 3.72564579502286   occ= 0 | 0
```

Part I. Basic DFT Calculation

<Output>

```
 ** Mulliken pop on meta-lowdin orthogonal AOs  **
 ** Mulliken pop  **
pop of  0 O 1s          1.99996
pop of  0 O 2s          1.64653
pop of  0 O 3s          0.00367
pop of  0 O 2px         1.99177
pop of  0 O 2py         1.29611
pop of  0 O 2pz         1.65644
pop of  0 O 3px         0.00228
pop of  0 O 3py         0.00940
pop of  0 O 3pz         0.00137
pop of  0 O 3dxy        0.00000
pop of  0 O 3dyz        0.00435
pop of  0 O 3dz^2       0.00159
pop of  0 O 3dxz        0.00126
pop of  0 O 3dx2-y2     0.00159
pop of  1 H 1s          0.67570
pop of  1 H 2s          0.01110
pop of  1 H 2px         0.00235
pop of  1 H 2py         0.00131
pop of  1 H 2pz         0.00139
pop of  2 H 1s          0.67570
pop of  2 H 2s          0.01110
pop of  2 H 2px         0.00235
pop of  2 H 2py         0.00131
pop of  2 H 2pz         0.00139
 ** Mulliken atomic charges  **
charge of    0O =     -0.61632
charge of    1H =      0.30816
charge of    2H =      0.30816
```

$\delta^- \approx -0.62$



$\delta^+ \approx 0.31$ $\delta^+ \approx 0.31$

B3LYP/ccpvdz

Population analysis

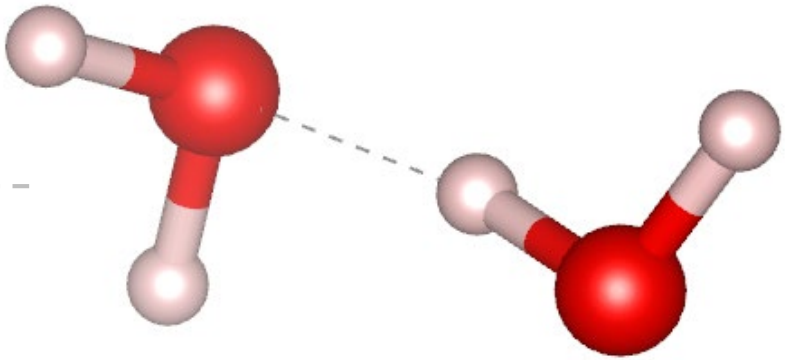
→ Mulliken charge (LCAO-MO)

→ Bader charge, ...

Part I. Basic DFT Calculation

Exercise 1. Water dimer interaction energy

	Ele.	X	Y	Z
	O	0.000	0.000	0.000
Mon1	H	-0.635	0.628	0.391
	H	0.869	0.421	0.090
	O	-0.277	0.479	-2.567
Mon2	H	-0.975	-0.048	-2.956
	H	-0.230	0.212	-1.603



* Interaction energy

E(AB) – E(A) – E(B)

B3LYP/ccpvdz

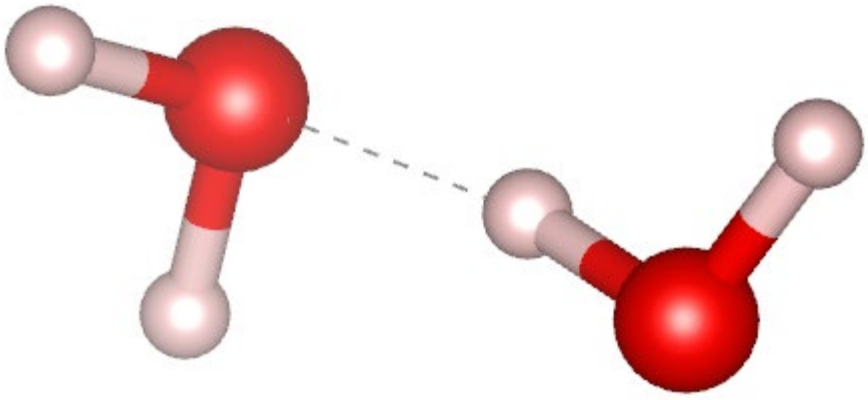
Result) **-6.810** *kcal/mol*

```
e = mydft.kernel()
e1 = mydft2.kernel()
e2 = mydft3.kernel()
print((e - e1 - e2)*627.509)
```

* 1 *Hartree* = 627.509 *kcal/mol*

Part I. Basic DFT Calculation

Exercise 1. Water dimer interaction energy



	B3LYP	PBE
cc-pVDZ	-6.810	-7.985
cc-pVTZ	-4.469	-5.565
cc-pVQZ	-3.659	-4.757

Ref. -3.395 *kcal/mol*

DLPNO-CCSD(T)-F12/aug-cc-pVQZ

S. Song, et al., Nat. Commun. 14, 799 (2023)

Part I. Basic DFT Calculation

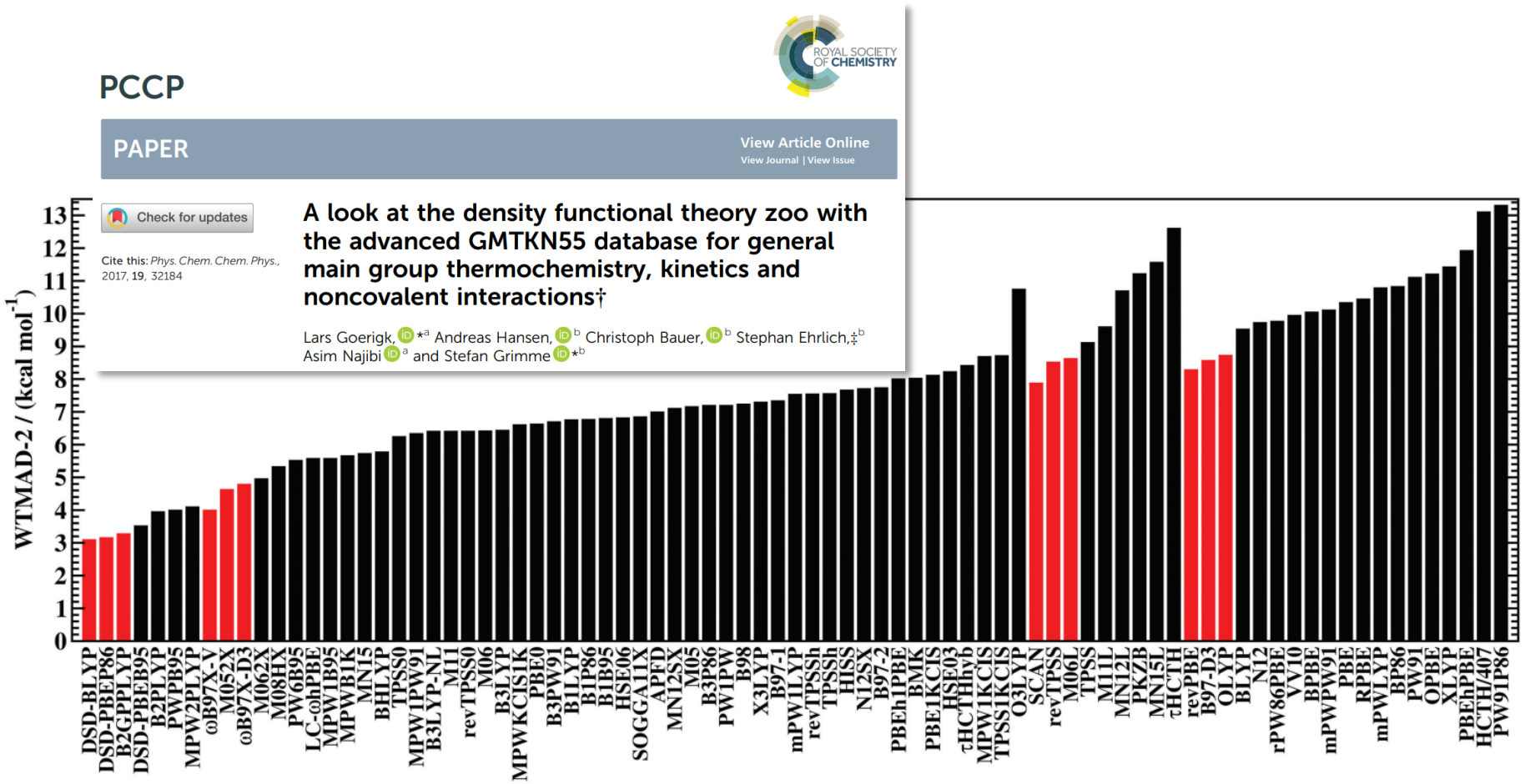


Fig. 8 Final WTMAD-2 values over the entire GMTKN55 for all assessed 83 dispersion-corrected DFAs (kcal mol^{-1}). The red bars indicate the three best approaches on their respective rung of Jacob's Ladder. The suffix "D3" was omitted in all cases, unless it is needed to avoid ambiguity.

- Comprehensive insight into {functional/basis sets} is required.

L. Goerigk, et al., Phys. Chem. Chem. Phys. 19, 32184-32215 (2017)

Part II. Geometry Optimization

** pip install pyberny (geometry optimizer)

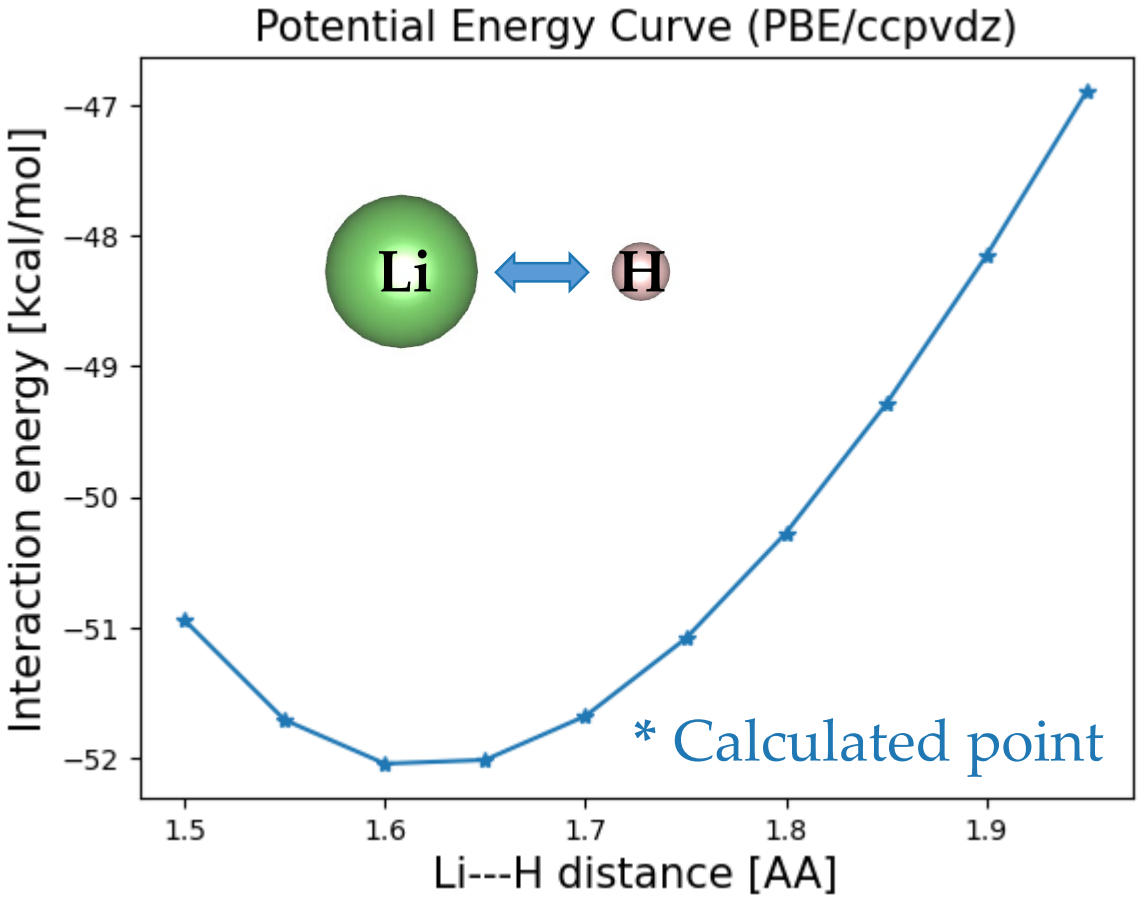
```
pip install pyberny
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Collecting pyberny
  Downloading pyberny-0.6.3-py3-none-any.whl (27 kB)
Requirement already satisfied: numpy<2.0,>=1.15 in /usr/local/lib/python3.9/dist-packages (from pyberny) (1.22.4)
Installing collected packages: pyberny
Successfully installed pyberny-0.6.3
```

<Input code>

```
from pyscf import gto, dft
from pyscf.geomopt.berny_solver import optimize

mol = gto.M(atom='Li 0 0 0; H 0 0 1.5', basis='ccpvdz', verbose = 4)
mydft = dft.RKS(mol)
mydft.xc = 'pbe'
mol_eq = optimize(mydft)
print(mol_eq.atom_coords()*0.529) # 1bohr = 0.529 Angstrom
```

Part II. Geometry Optimization



- Finding optimal(most stable) structure for a given system {charge, spin state, ...}

Part II. Geometry Optimization

<Output>

```
INFO:pyscf.geomopt.berny_solver.140709643640160:5 * All criteria matched
[[ 0.          0.         -0.06005129]  →X Y Z (Li)
 [ 0.          0.          1.55954897]] →X Y Z (H)
```

```
    print(mol_eq.atom_coords()*0.529) # 1bohr = 0.529 Angstrom
```

⇒ Optimal distance Li-H (PBE/ccpvdz) : **1.62** Å

** verbose = 4

```
Geometry optimization cycle 4
Cartesian coordinates (Angstrom)
 Atom          New coordinates                dX          dY          dZ
  Li    0.000000   0.000000  -0.058466    0.000000  0.000000 -0.008572
   H    0.000000   0.000000   1.558466    0.000000  0.000000  0.008572
```

Part II. Geometry Optimization

Database <Cluster system>

Computational Chemistry Comparison and Benchmark DataBase Release 22 (May 2022) Standard Reference Database 101 National Institute of Standards and Technology

Home | Experimental | Calculated | Comparisons | Resources | FAQ Help

You are here: **Home**

The CCCBDB contains:

Experimental and computed (quantum mechanics) thermochemical data for a selected set of 2186 **gas-phase** atoms and small molecules.
Tools for comparing experimental and computational ideal-gas thermochemical properties.
Vibrational Frequencies, Rotational Constants, Electric Dipole, Electric Quadrupole, Polarizabilities

Molecules in the CCCBDB mostly have the following constraints:

- Well-established experimental heat of formation.
- Atoms with atomic number less than 36 (Krypton) with only a few transition metals. We have added a few molecules containing Te, I, and Xe
- Less than 15 heavy atoms and less than 30 atoms total. Except for a few larger molecules: tetracene, triphenylmethane, coronene, and C60.

Citation	NIST Computational Chemistry Comparison and Benchmark Database NIST Standard Reference Database Number 101 Release 22, May 2022, Editor: Russell D. Johnson III http://cccbdb.nist.gov/ DOI:10.18434/T47C7Z

NIST policy on privacy, security, and accessibility.

© 2015 copyright by the U.S. Secretary of Commerce on behalf of the United States of America. All rights reserved.

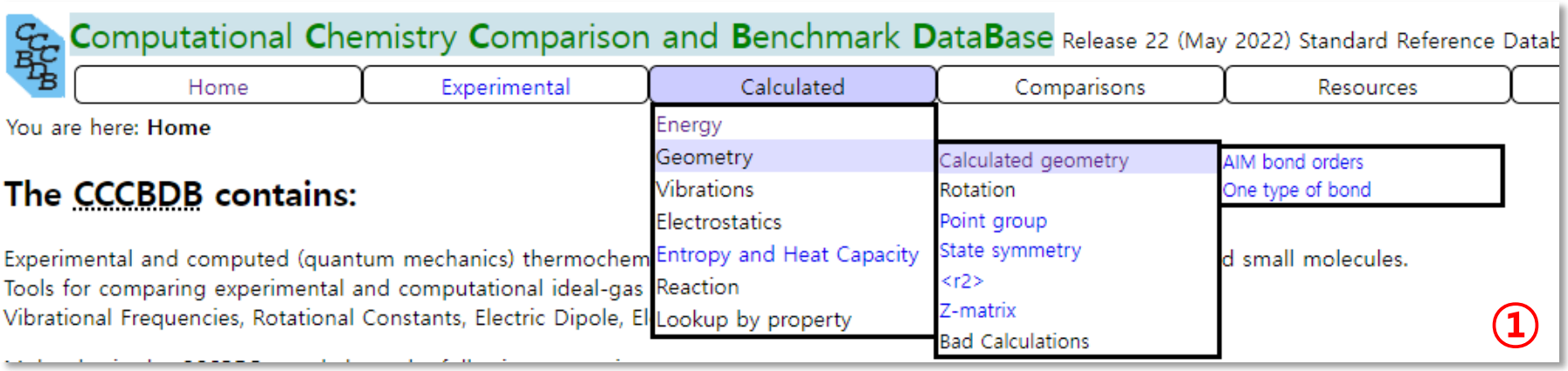
The National Institute of Standards and Technology (NIST) is an agency of the U.S. Department of Commerce.

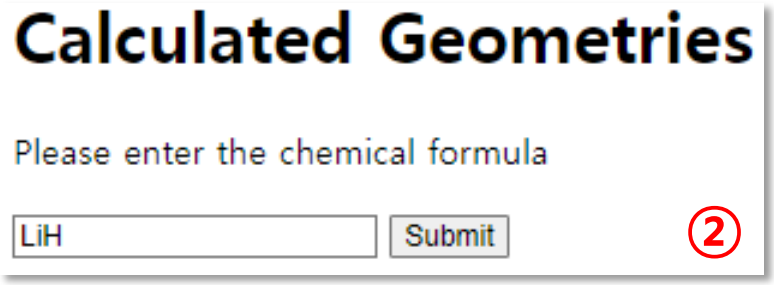
Please send questions, comments, corrections, additions and suggestions to cccbdb@nist.gov.

NIST
National Institute of Standards and Technology
U.S. Department of Commerce

Part II. Geometry Optimization

Database <Cluster system>





Part II. Geometry Optimization

Database <Cluster system>

Basis sets

Methods

Methods with standard basis sets

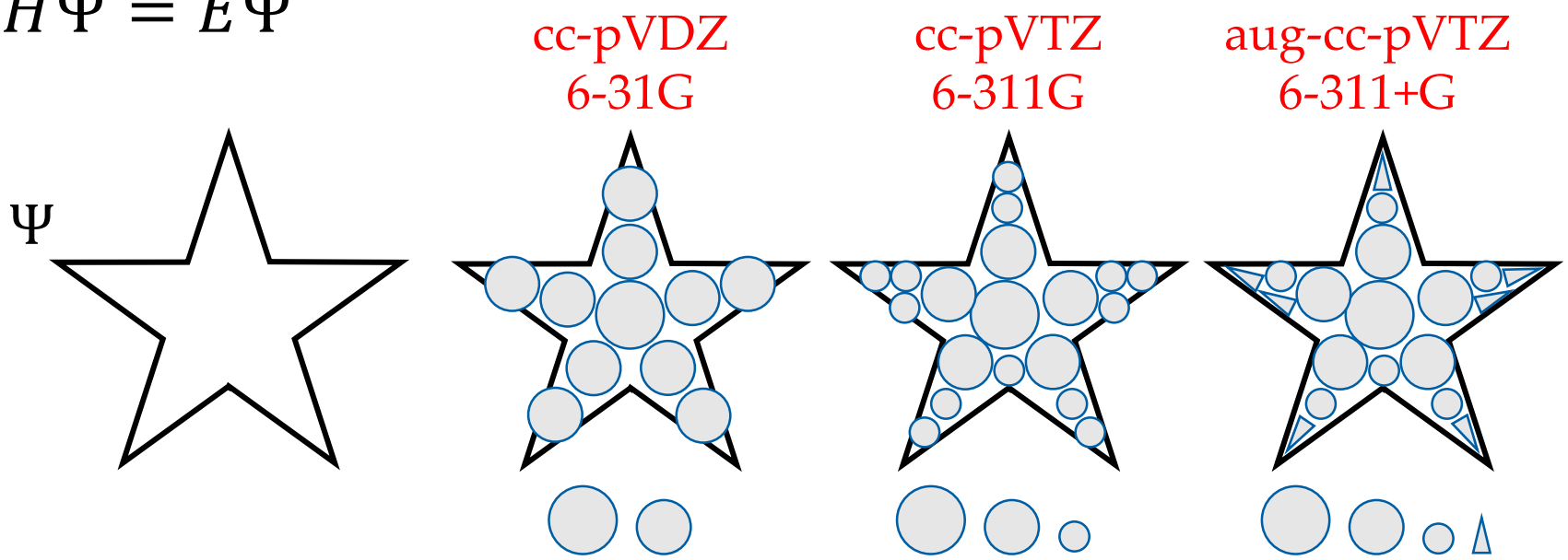
		STO-3G	3-21G	3-21G*	6-31G	6-31G*	6-31G**	6-31+G**	6-311G*	6-311G**	6-31G(2df,p)	6-311+G(3df,2p)	6-311+G(3df,2pd)	TZVP	cc-pVDZ	cc-pVTZ
hartree fock	HF	1.511	1.640	1.640	1.640	1.636	1.630	1.629	1.608	1.607	1.628	1.608	1.608	1.612	1.619	1.608
density functional	LSDA	1.534	1.646	1.646	1.641	1.636	1.629	1.627	1.605	1.602	1.624	1.599	1.597	1.609	1.615	1.599
	BLYP	1.537	1.638	1.638	1.632	1.630	1.622	1.621	1.603	1.600	1.621	1.597	1.596	1.607	1.611	1.597
	B1B95	1.530	1.636	1.636	1.633	1.628	dnf	1.621	1.599	1.599	1.621	1.597	1.596	1.605	1.611	1.596
	B3LYP	1.526	1.630	1.630	1.625	1.621	1.615	1.614	1.595	1.592	1.615	1.590	1.590	1.599	1.604	1.590
	B3LYPultrafine	1.526	1.630	1.630	1.625	1.621	1.615	1.614	1.595	1.592	1.615	1.591	1.590	1.600	1.604	1.590
	B3PW91	1.531	1.639	1.639	1.636	1.631	1.625	1.624	1.603	1.600	1.624	1.599	1.599	1.607	1.613	1.599
	mPW1PW91	1.529	1.638	1.638	1.636	1.630	1.624	1.623	1.602	1.599	1.624	1.598	1.597	1.606	1.613	
	M06-2X	1.520	1.620	1.620	1.623	1.620	1.615	1.614	1.591	1.590	1.619	1.588	1.587	1.596	1.604	1.600
	PBEPBE	1.542	1.648			1.637	1.632	1.631	1.610	1.607	1.631	1.605	1.604	1.615	1.620	1.605
	PBEPBEultrafine	1.542	1.648	1.648	1.643	1.637	1.632	1.631	1.611	1.607	1.631	1.605	1.604	1.615	1.620	1.605
	PBE1PBE	1.530	1.639	1.639	1.636	1.631	1.625	1.624	1.603	1.600	1.624	1.599	1.598	1.607	1.613	1.599
	HSEh1PBE	1.530	1.638	1.638	1.635	1.631	1.624	1.623	1.603	1.600	1.624	1.599	1.598	1.607	1.613	1.598
	TPSSh					1.631		1.626			1.627					1.606
	wB97X-D			1.649		1.643		1.637		1.623				1.626	1.637	1.622
	B97D3		1.656			1.647		1.640		1.618		1.616		1.626		1.615
⋮	⋮									⋮						
Coupled Cluster	CCD	1.546	1.669	1.669	1.667	1.646	1.627	1.626	1.615	1.598	1.628	1.609	1.607	1.604	1.617	1.607
	CCSD	1.547	1.673	1.673	1.672	1.649	1.630	1.629	1.618	1.600	1.631	1.610	1.608	1.607	1.619	1.615
	CCSD=FULL	1.547	1.673	1.673	1.672	1.648	1.629	1.627	1.616	1.595	1.624	1.600	1.598	1.605	1.615	1.607
	CCSD(T)	1.547	1.673	1.674	1.672	1.654	1.630	1.629	1.618	1.600	1.630	1.610	1.608	1.607	1.619	1.608
	CCSD(T)=FULL	1.547	1.674	1.674	1.672	1.652	1.629	1.628	1.615	1.595	1.620	1.600	1.597	1.605	1.615	1.596
		STO-3G	3-21G	3-21G*	6-31G	6-31G*	6-31G**	6-31+G**	6-311G*	6-311G**	6-31G(2df,p)	6-311+G(3df,2p)	6-311+G(3df,2pd)	TZVP	cc-pVDZ	cc-pVTZ

General tips)

- The rightmost one with a similar name(basis sets)
- e.g.) B3LYP or CCSD(T) / aug-cc-pvqz

* Basis sets

$\hat{H}\Psi = E\Psi$



Gaussian-type orbital, GTO

(Basis functions)

* Dunning family) *cc-pVnZ*, *n*=*D*,*T*,*Q*,5,...

Pople) *6-31G* + {*, +, ()}

cc-pVDZ	cc-pVTZ	cc-pVQZ	aug-cc-pVDZ	aug-cc-pVTZ

STO-3G	3-21G	3-21G*	6-31G	6-31G*	6-31G**

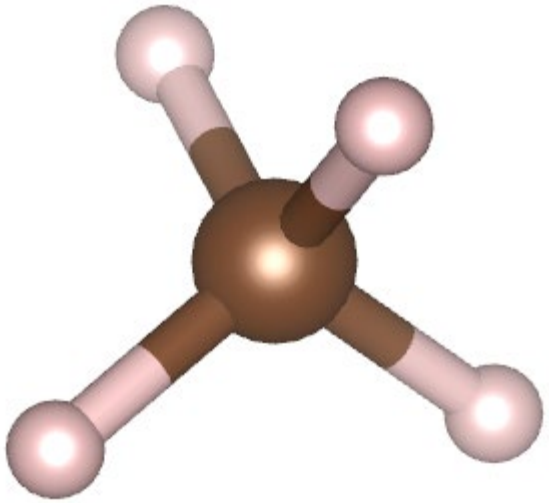
Part II. Geometry Optimization

Exercise 2. CH_3OH(Methanol) optimization

Suppose CH_3OH structure is unknown. (can not find in database.)

Strategy) Replace one H with OH in the CH_4 structure.

CH_4	X	Y	Z
C	0.0000	0.0000	0.0000
H	0.6386	0.6386	0.6386
H	-0.6386	-0.6386	0.6386
H	-0.6386	0.6386	-0.6386
H	0.6386	-0.6386	-0.6386



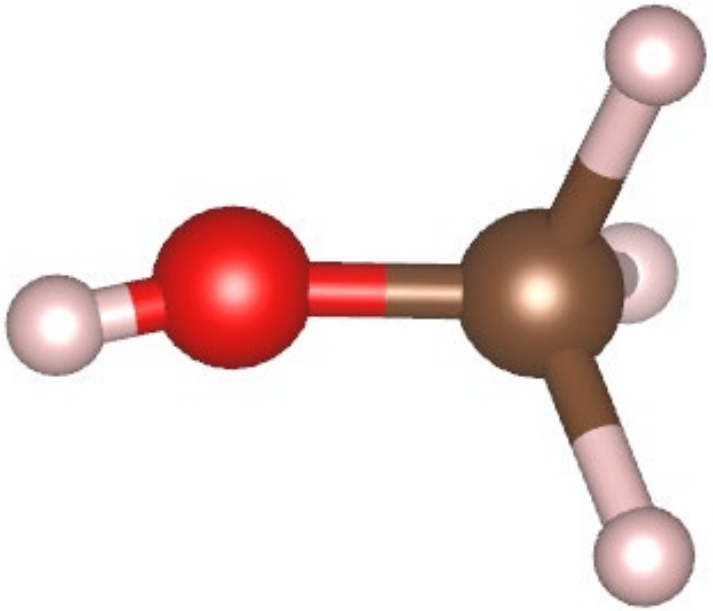
Part II. Geometry Optimization

Exercise 2. CH_3OH(Methanol) optimization

Suppose CH_3OH structure is unknown. (can not find in database.)

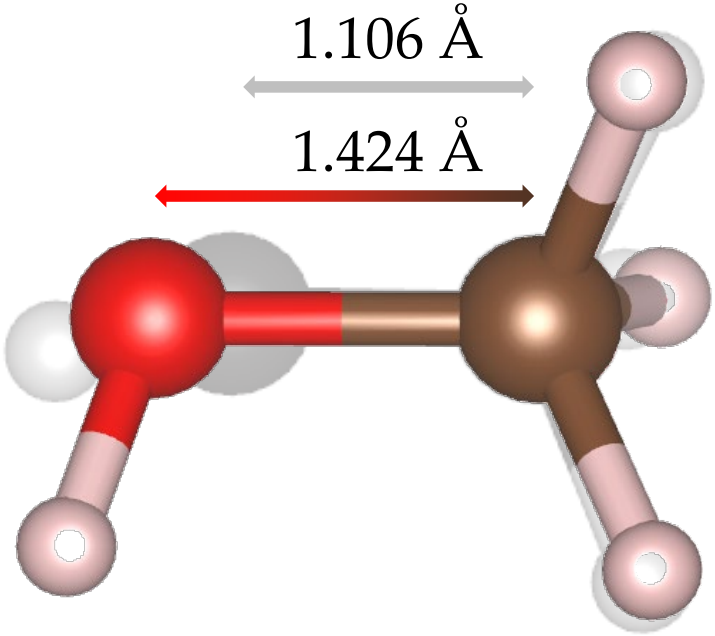
Strategy) Replace one H with OH in the CH_4 structure.

CH_3OH (ini. gue.)	X	Y	Z
C	0.0000	0.0000	0.0000
H	0.6386	0.6386	0.6386
H	-0.6386	-0.6386	0.6386
H	-0.6386	0.6386	-0.6386
O	0.6386	-0.6386	-0.6386
H	**1.1386**	**-0.9386**	**-0.9386**

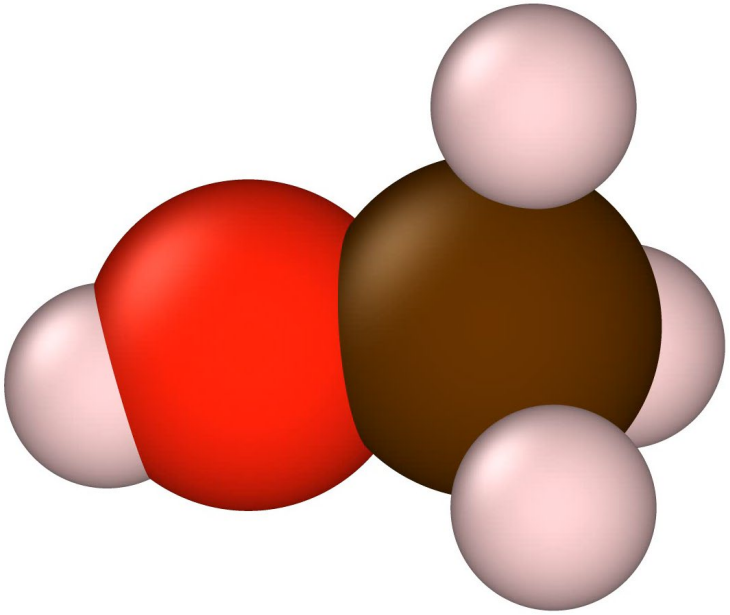


Part II. Geometry Optimization

Exercise 2. CH_3OH(Methanol) optimization



<VESTA>



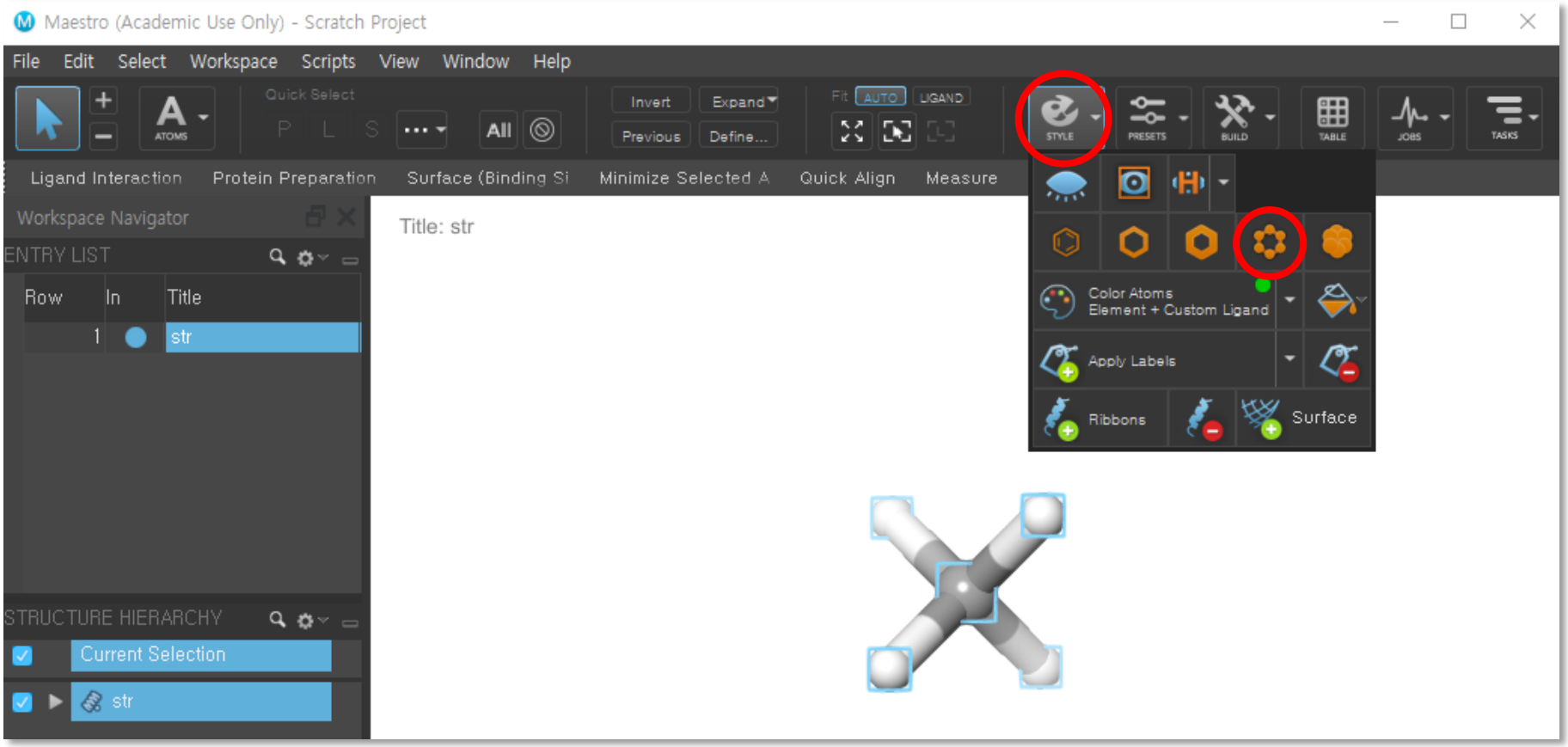
<OVITO>

A. Stukowski, Modelling Simul. Mater. Sci. Eng. 18, 015012 (2010)

Part II. Geometry Optimization

Exercise 2. CH_3OH(Methanol) optimization

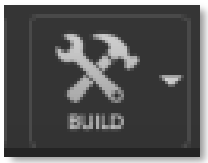
** Generating a reasonable initial guess structure (Maestro)

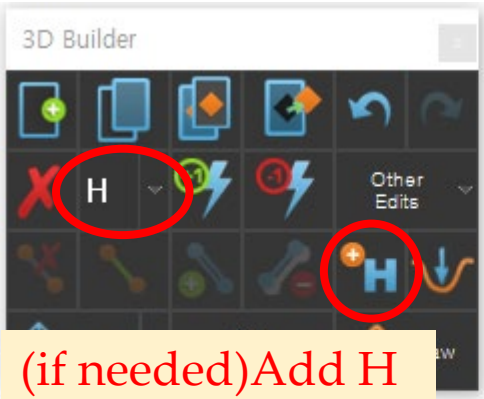


Part II. Geometry Optimization

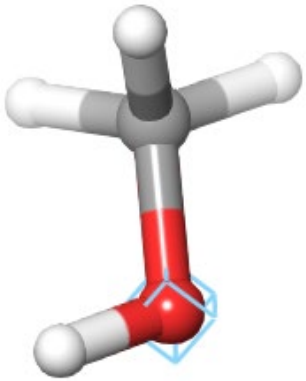
Exercise 2. CH_3OH(Methanol) optimization

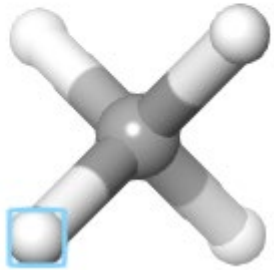
** Generating a reasonable initial guess structure (Maestro)



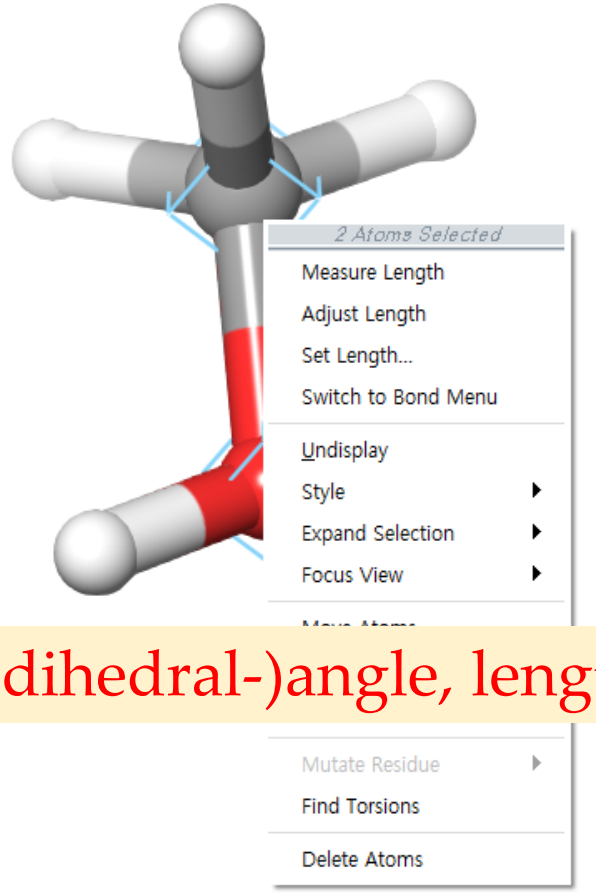


(if needed)Add H





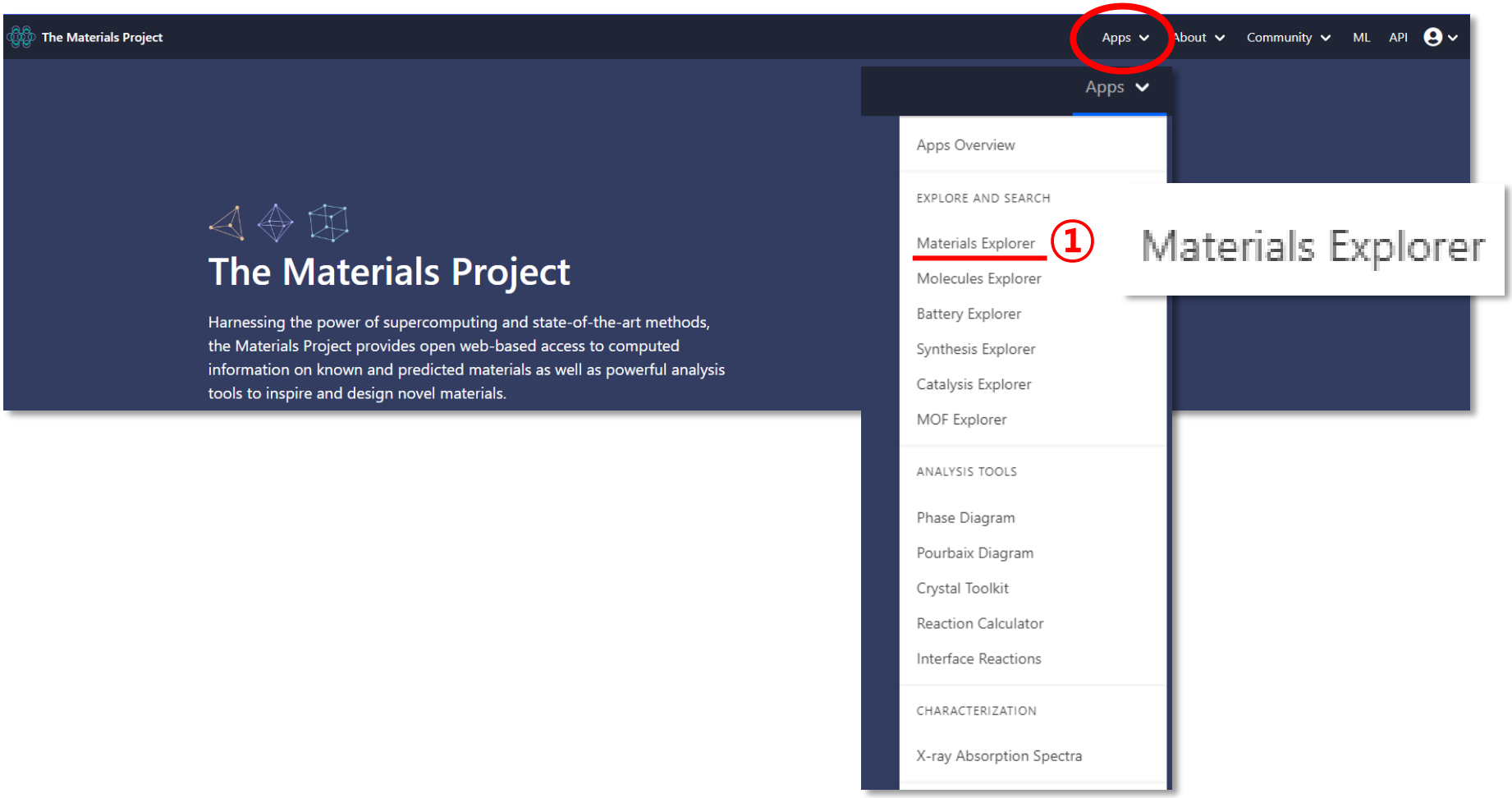
Select H



Set (dihedral-)angle, length

Part II. Geometry Optimization

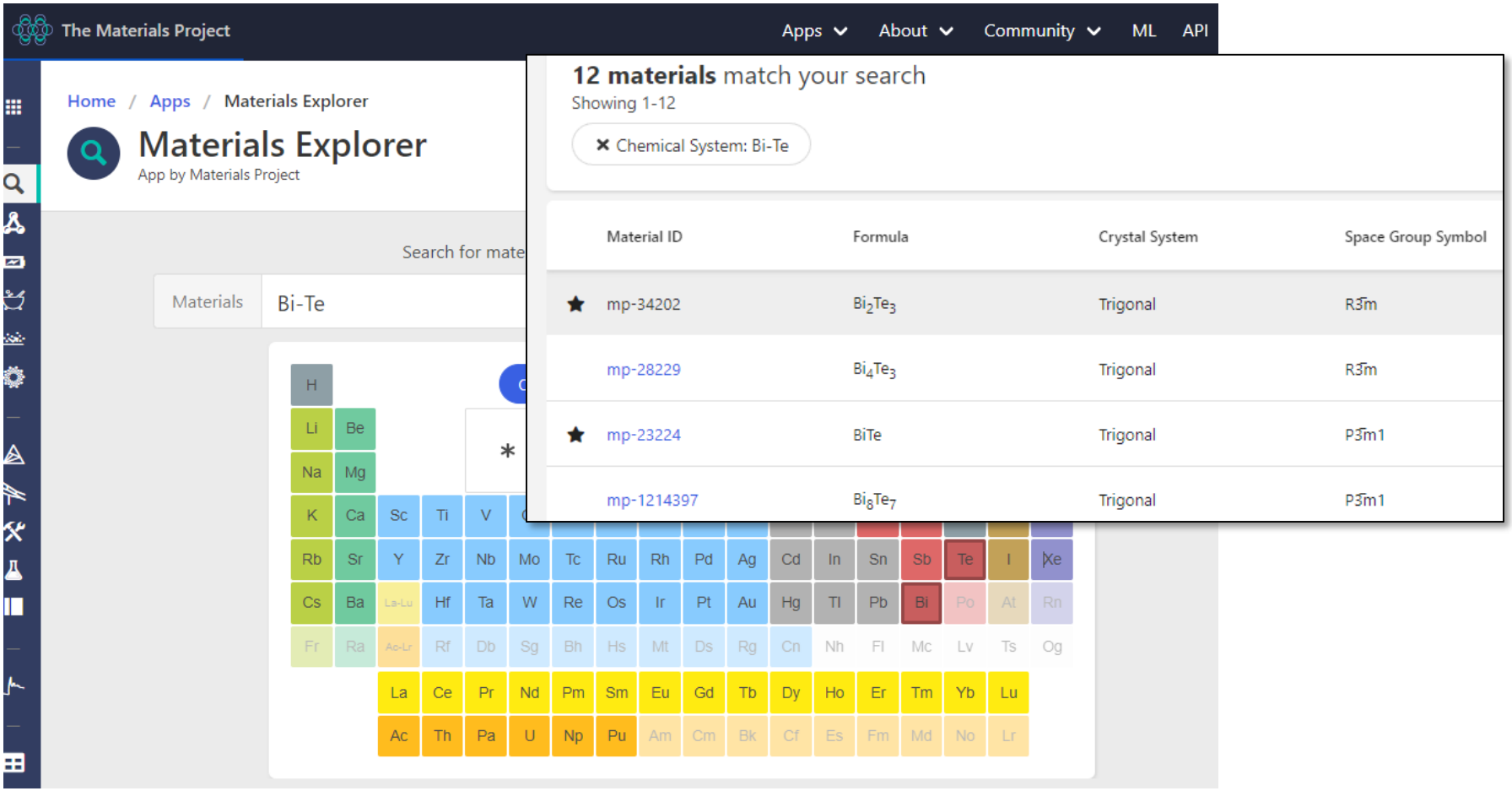
Database <Periodic system>



Part II. Geometry Optimization

Database <Periodic system>

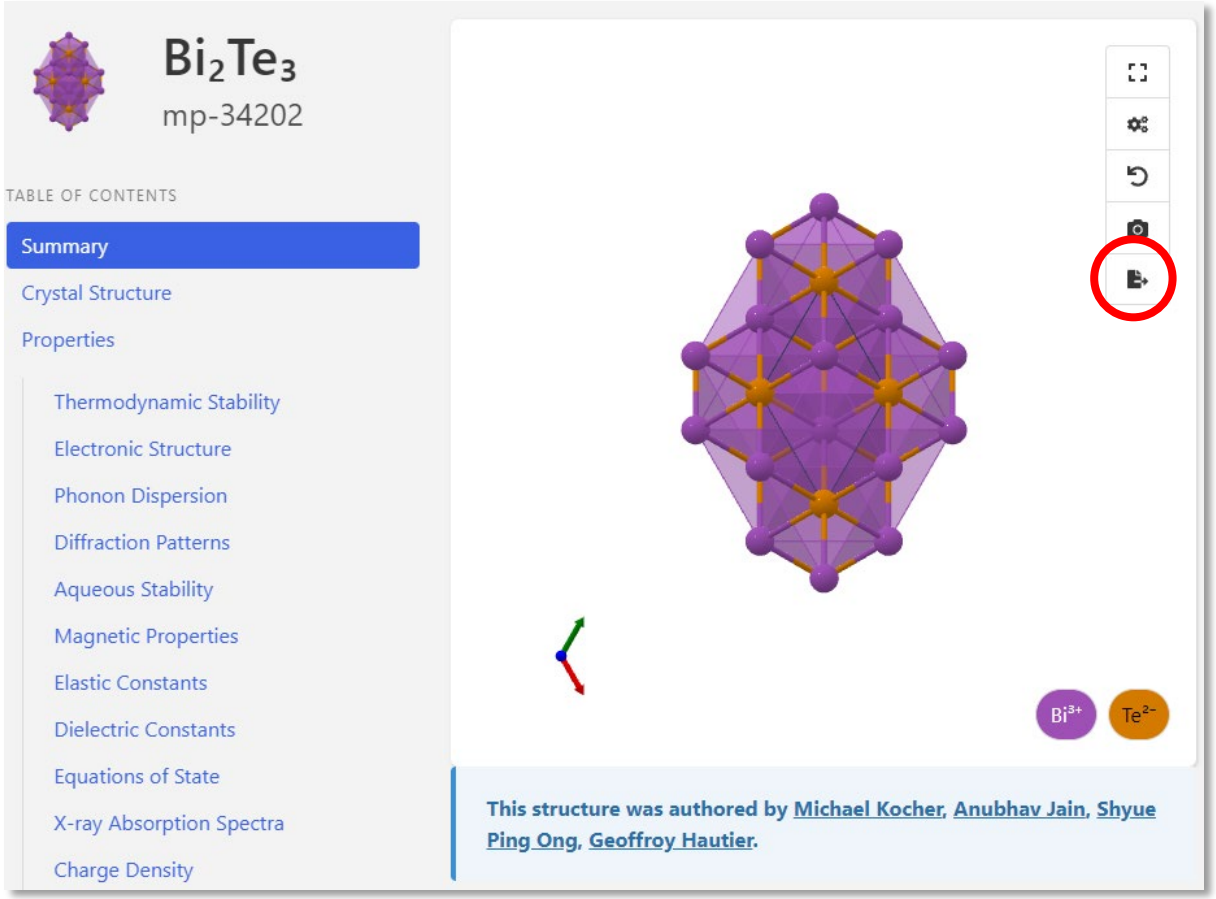
e.g.) Bi_2Te_3 unitcell structure

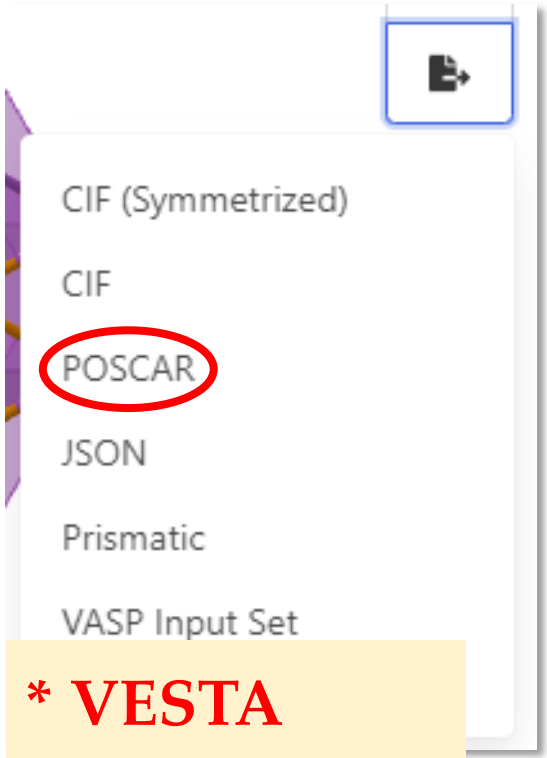


Part II. Geometry Optimization

Database <Periodic system>

e.g.) Bi_2Te_3 unitcell structure





*** VESTA**

Bi2Te3.poscar

→ Bi2Te3.vasp

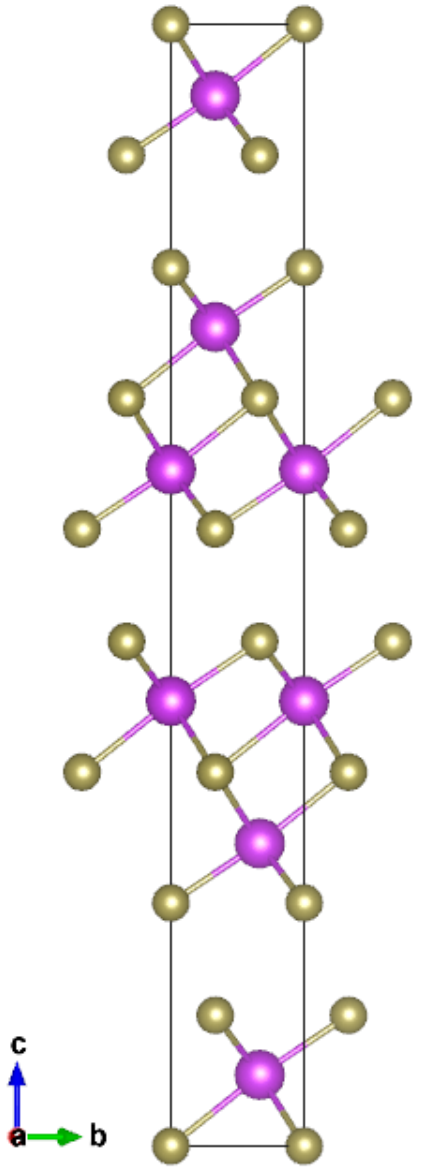
Part II. Geometry Optimization

Database <Periodic system>

e.g.) Bi_2Te_3 unitcell structure

```
Bi6 Te9
1.0
   2.2090652234007724    -3.8262132041636301
   2.2090652234007724     3.8262132041636301
   0.0000000000000000     0.0000000000000000
Bi Te
6 9
direct
   0.3333333333333333     0.6666666666666666
   0.3333333333333333     0.6666666666666667
   0.0000000000000000     0.0000000000000000
   0.0000000000000000     0.0000000000000000
   0.6666666666666666     0.3333333333333333
   0.6666666666666666     0.3333333333333335
   0.6666666666666666     0.3333333333333333
   0.0000000000000000     0.0000000000000000
   0.0000000000000000     0.0000000000000000   0.2168077399999997  Te2-
   0.3333333333333333     0.6666666666666666   0.4498589266666669  Te2-
   0.6666666666666666     0.3333333333333333   0.3333333333333333  Te2-
   0.6666666666666666     0.3333333333333333   0.5501410733333331  Te2-
   0.0000000000000000     0.0000000000000000   0.7831922600000002  Te2-
   0.3333333333333333     0.6666666666666666   0.6666666666666666  Te2-
   0.3333333333333333     0.6666666666666666   0.8834744066666664  Te2-
```

* POSCAR format
Comment line
Scaling factor
Lattice vectors
Atoms
Number of atoms
Positions



Part III. Dispersion Correction

(Grimme) DFT-D4

$$E_{total} = E_{DFA} + E_{disp}$$ E_{DFA} : Part I

* Lots of dispersion correction methods. (XDM, MBD, vv10, ...)

⇒ DFT-D4 linked with PySCF

```
pip install dftd4

Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Collecting dftd4
  Downloading dftd4-3.5.0-cp39-cp39-manylinux_2_12_x86_64.manylinux2010_x86_64.whl (15.2 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 15.2/15.2 MB 11.0 MB/s eta 0:00:00
Requirement already satisfied: numpy in /usr/local/lib/python3.9/dist-packages (from dftd4) (1.22.4)
Requirement already satisfied: cffi in /usr/local/lib/python3.9/dist-packages (from dftd4) (1.15.1)
Requirement already satisfied: pycparser in /usr/local/lib/python3.9/dist-packages (from cffi->dftd4) (2.21)
Installing collected packages: dftd4
Successfully installed dftd4-3.5.0
```


E. Caldeweyher, et al., J. Chem. Phys. 150, 154122 (2019)

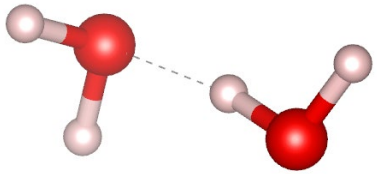
Part III. Dispersion Correction

<Input code>

Added line

```
import dftd4.pyscf as disp
from pyscf import gto, dft

bas='ccpvdz'
xc = 'pbe'

dim = gto.M(
    atom = '''
    O 0.000 0.000 0.000
    H -0.635 0.628 0.391
    H 0.869 0.421 0.090
    O -0.277 0.479 -2.567
    H -0.975 -0.048 -2.956
    H -0.230 0.212 -1.603''',
    spin=0,
    charge=0,
    basis = bas,
    verbose=0)
```

```
mydft = dft.RKS(dim)
mydft.xc = xc
e_dfa = mydft.kernel()

d4 = disp.DFTD4Dispersion(mol,xc=xc)
e_d4 = d4.kernel()[0]

e_dimer = e_dfa + e_d4
```



* Interaction energy

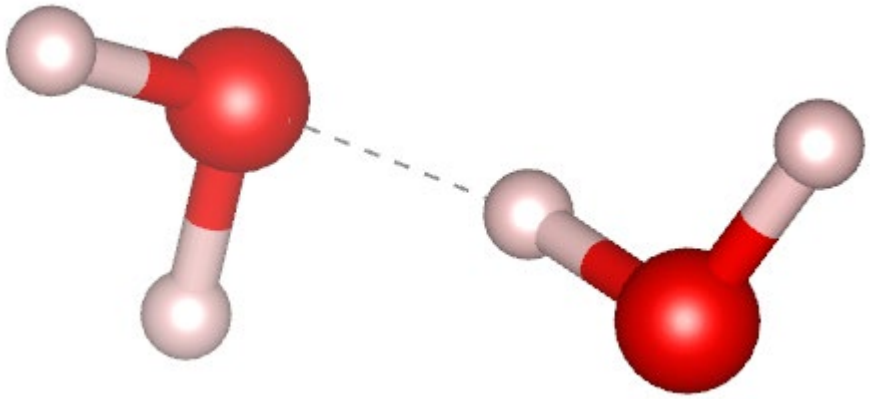
$$E_{int} = E_{dimer} - E_{mon1} - E_{mon2}$$

Part I. $E_{sys} = E_{DFA(B3LYP, PBE\ldots)}$

Part III. $E_{sys} = E_{DFA} + \boldsymbol{E_{disp}}$

Part III. Dispersion Correction

Exercise 1. Water dimer interaction energy



Ref. -3.395 *kcal/mol*

DLPNO-CCSD(T)-F12/aug-cc-pVQZ

	B3LYP	B3LYP-D4	PBE	PBE-D4
cc-pVDZ	-6.810	-7.501	-7.985	-8.427
cc-pVTZ	-4.469	-5.160	-5.565	-6.007
cc-pVQZ	-3.659	**-4.350**	-4.757	**-5.199**

- (D4) Independent on basis sets
- Mostly improved results (exceptions of course exist.) right here..

PBE-D4 value corrected (23.05.01)

Summary

Part I. Basic DFT Calculation

- Make sure the input is well defined as the user intended.
- Comprehensive insight into {functional/basis sets} is required.

Part II. Geometry Optimization

- Initially, refer to the published database.
- If the desired structure does not exist,
  initial structure can be created from similar structures.

Part III. Dispersion Correction

- Almost all of the dispersion corrected DFT calculations
  give improved results.

This file is uploaded to tccl.yonsei.ac.kr
and the copyright belongs to TCCL