Tutorial of OpenMX

- About OpenMX
- Implementation of OpenMX
- Tutorial of OpenMX

Taisuke Ozaki
ISSP, Univ. of Tokyo

May, 2023

OpenMX Open source package for Material eXplorer

- Software package for density functional calculations of molecules and bulks
- Norm-conserving pseudopotentials (PPs)
- Variationally optimized numerical atomic basis functions

Basic functionalities

- SCF calc. by LDA, GGA, DFT+U
- Total energy and forces on atoms
- Band dispersion and density of states
- Geometry optimization by BFGS, RF, EF
- Charge analysis by Mullken, Voronoi, ESP
- Molecular dynamics with NEV and NVT ensembles
- Charge doping
- Fermi surface
- Analysis of charge, spin, potentials by cube files
- Database of optimized PPs and basis funcitons

Extensions

- O(N) and low-order scaling diagonalization
- Non-collinear DFT for non-collinear magnetism
- Spin-orbit coupling included self-consistently
- Electronic transport by non-equilibrium Green function
- Electronic polarization by the Berry phase formalism
- Maximally localized Wannier functions
- Effective screening medium method for biased system
- Reaction path search by the NEB method
- Band unfolding method
- STM image by the Tersoff-Hamann method
- etc.

History of OpenMX

2000 Start of development

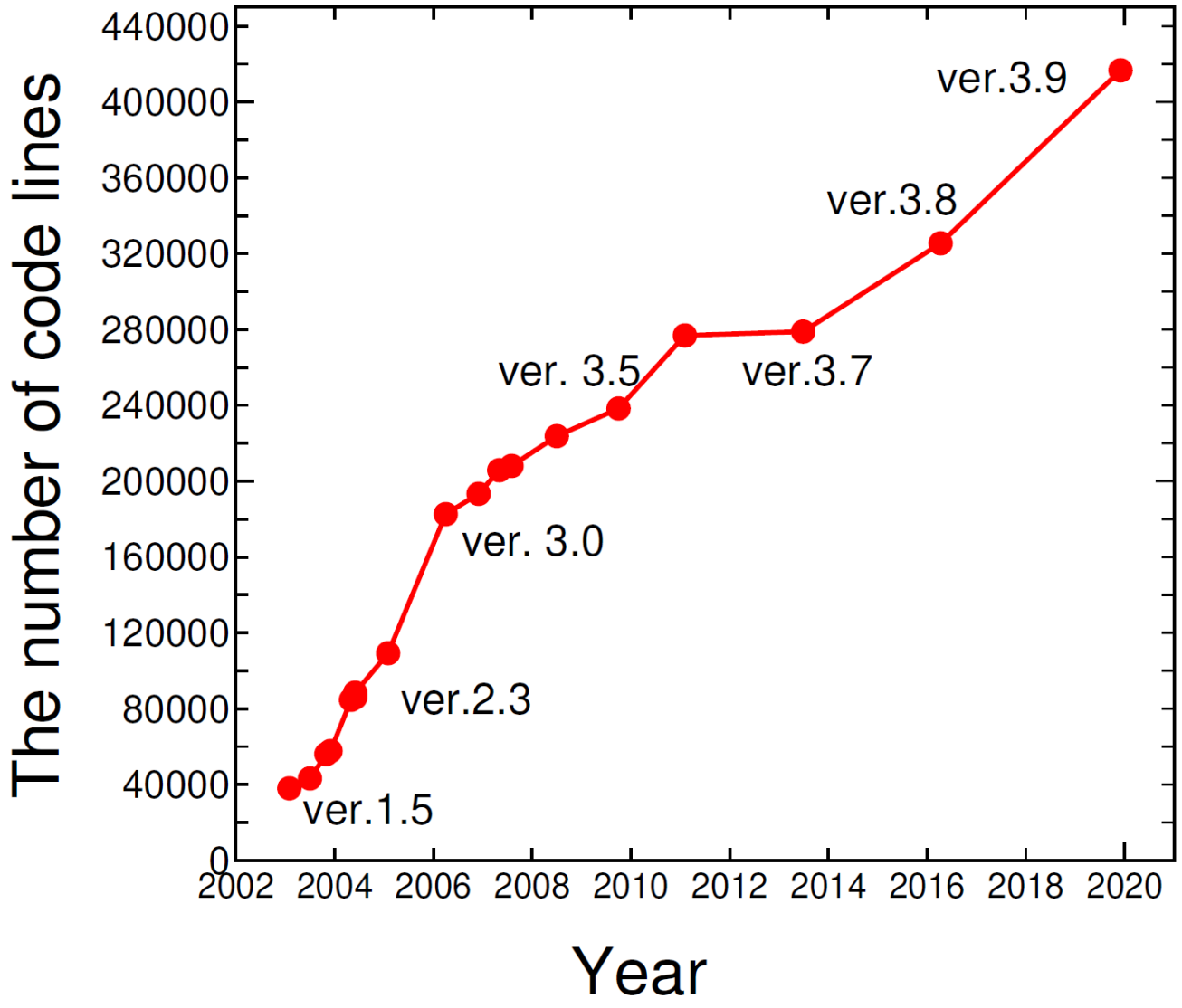
2003 Public release (GNU-GPL)

2003 Collaboration:
AIST, NIMS, SNU
KAIST, JAIST,
Kanazawa Univ.
CAS, UAM
NISSAN, Fujitsu Labs.
etc.

2019 19 public releases
Latest version: 3.9

http://www.openmx-square.org

Development of OpenMX code



Contributors to OpenMX development

T. Ozaki (Univ. of Tokyo)
H. Kino (NIMS)
J. Yu (SNU)
M.J. Han (KAIST)
M. Ohfuchi (Fujitsu Labs.)
F. Ishii (Kanazawa Univ.)
K. Sawada (Kanazawa Univ.)
Y. Kubota (Kanazawa Univ.)
Y.P. Mizuta (Kanazawa Univ.)
H. Kotaka (Kyoto Univ.)
N. Yamaguchi (Kanazawa Univ.)
H. Sawahata (Kanazawa Univ.)
T.B. Prayitno (Kanazawa Univ.)
T. Ohwaki (NISSAN ARC)
T.V.T Duy (AISIN SEIKI)
M. Miyata (JAIST)
G. Jiang (Wuhan Univ. of Sci.&Tech.)
T. Iitaka (RIKEN)

P.-H. Chang (George Mason Univ.)
A. Terasawa (TIT)
Y. Gohda (TIT)
H. Weng (CAS)
Y. Shiihara (Toyota Tech. Inst.)
M. Toyoda (Tokyo Inst. Tech.)
Y. Okuno (FUJIFILM)
R. Perez (UAM)
P.P. Bell (UAM)
M. Ellner (UAM)
Yang Xiao (NUAA)
A.M. Ito (NIFS)
M. Otani (AIST)
M. Kawamura (Univ. of Tokyo)
K. Yoshimi (Univ. of Tokyo)
C.-C. Lee (Tamkang Univ.)
Y.-T. Lee (Academia Sinica)
M. Fukuda (Univ. of Tokyo)
S. Ryee (KAIST)
K. Terakura (AIST)

Materials studied by OpenMX

First characterization of silicene on ZrB_2 in collaboration with experimental groups

A. Fleurence et al., Phys. Rev. Lett. 108, 245501 (2012).

First identification of Jeff=1/2 Mott state of Ir oxides

B.J. Kim et al., Phys. Rev. Lett. 101, 076402 (2008).

Theoretical proposal of topological insulators

C.-H. Kim et al., Phys. Rev. Lett. 108, 106401 (2012).
H. Weng et al., Phy. Rev. X 4, 011002 (2014).

First-principles molecular dynamics simulations for Li ion battery

T. Ohwaki et al., J. Chem. Phys. 136, 134101 (2012).
T. Ohwaki et al., J. Chem. Phys. 140, 244105 (2014).

Magnetic anisotropy energy of magnets

Z. Torbatian et al., Appl. Phys. Lett. 104, 242403 (2014).
I. Kitagawa et al., Phys. Rev. B 81, 214408 (2010).

Electronic transport of graphene nanoribbon on surface oxidized Si

H. Jippo et al., Appl. Phys. Express 7, 025101 (2014).
M. Ohfuchi et al., Appl. Phys. Express 4, 095101 (2011).

Interface structures of carbide precipitate in bcc-Fe

H. Sawada et al., Modelling Simul. Mater. Sci. Eng. 21, 045012 (2013).

Universality of medium range ordered structure in amorphous metal oxides

K. Nishio et al., Phys. Rev. Lett. 340, 155502 (2013).

Materials treated so far

Silicene, graphene
Carbon nanotubes
Transition metal oxides
Topological insulators
Intermetallic compounds
Molecular magnets
Rare earth magnets
Lithium ion related materials
Structural materials
etc.

About 1200 published papers

Information of OpenMX

http://www.openmx-square.org/index.html

Welcome to OpenMX

Open source package for Material eXplorer

- What's new
  - Patch3.9.2 to OpenMX Ver 3.9 (Feb. 11, 2020)
  - Patch3.9.1 to OpenMX Ver 3.9 (Jan. 02, 2020)

- What is OpenMX?
- Download
- Manual of Ver. 3.9
- Manual of Ver. 3.8
- Technical Notes
- Video Lectures
- Publications
- OpenMX Forum
- OpenMX Viewer
- Workshop
- Database of Results
- Database of VPS and PAO
  - Ver. 2019
  - Ver. 2019 for core excitations
- ADPACK
- Miscellaneous informations
- Contributors
- Acknowledgment
- Opening positions
- Links

- Source codes
- Manual (Eng./日本語)
- OpenMX Forum
- Technical notes
- Publications
- Database of basis functions and pseudopotentials

Getting stated with OpenMX…

- **Install by yourself**
- **Use of preinstalled version in Computer centers**
- **Use of MateriApps Live!**

Let's start a simple calculation

Perform a SCF calculation of methane molecule.

```
ozaki@mx17:~/openmx3.9/work
[ozaki@mx17 work]$ mpirun -np 4 ./openmx Methane.dat | tee met.std
```

After finishing the calculation, several files are output as follows:

```
ozaki@mx17:~/openmx3.9/work
[ozaki@mx17 work]$ ls met.*
met.cif         met.ene  met.md2  met.std         met.v0.cube     met.xyz
met.dden.cube   met.md   met.out  met.tden.cube   met.vhart.cube
[ozaki@mx17 work]$
```

Input file of OpenMX

```
System.CurrrentDirectory         ./    # default=./
System.Name                      met
level.of.stdout                   1    # default=1 (1-3)
level.of.fileout                  1    # default=1 (0-2)

#
# Definition of Atomic Species
#

Species.Number       2
<Definition.of.Atomic.Species
 H    H5.0-s1          H_PBE13
 C    C5.0-s1p1        C_PBE13
Definition.of.Atomic.Species>

#
# Atoms
#

Atoms.Number         5
Atoms.SpeciesAndCoordinates.Unit   Ang # Ang|AU
<Atoms.SpeciesAndCoordinates
 1  C      0.000000     0.000000     0.000000     2.0  2.0
 2  H     -0.889981    -0.629312     0.000000     0.5  0.5
 3  H      0.000000     0.629312    -0.889981     0.5  0.5
 4  H      0.000000     0.629312     0.889981     0.5  0.5
 5  H      0.889981    -0.629312     0.000000     0.5  0.5
Atoms.SpeciesAndCoordinates>
```

(1) Specify a value after keyword
(2) Arbitrary order of keywords
(3) # by comment

Several keywords:

System name: files will be saved with the name

# of species in the system

Definition of species

# of atoms

Unit of atomic coordinate

Atomic coordinates

Initial occupation for up and down states of each atom

Output files

After finishing the calculation of ‘Methane.dat’, 11 files and one directory are generated.

met.std	standard output of the SCF calculation
met.out	input file and standard output
met.xyz	final geometrical structure
met.ene	values computed at every MD step
met.md	geometrical structures at every MD step
met.md2	geometrical structure of the final MD step
met.cif	cif file of the initial structure for Material Studio
met.tden.cube	total electron density in the Gaussian cube format
met.v0.cube	Kohn-Sham potential in the Gaussian cube format
met.vhart.cube	Hartree potential in the Gaussian cube format
met.dden.cube	difference electron density measured from atomic density
met_rst/	directory storing restart files

Contents of met.out

History of SCF

```
************************************************************
************************************************************
                SCF history at MD= 1
************************************************************
************************************************************

  SCF=   1  NormRD=  1.000000000000  Uele= -3.523143659974
  SCF=   2  NormRD=  0.567253699744  Uele= -4.405605131921
  SCF=   3  NormRD=  0.103433490729  Uele= -3.982266241934
  SCF=   4  NormRD=  0.024234990593  Uele= -3.906896836134
  SCF=   5  NormRD=  0.011006215697  Uele= -3.893084558820
  SCF=   6  NormRD=  0.006494145332  Uele= -3.890357113476
  SCF=   7  NormRD=  0.002722267527  Uele= -3.891669816209
  SCF=   8  NormRD=  0.000000672350  Uele= -3.889285164733
  SCF=   9  NormRD=  0.000000402419  Uele= -3.889285102456
  SCF=  10  NormRD=  0.000000346348  Uele= -3.889285101128
  SCF=  11  NormRD=  0.000000515395  Uele= -3.889285101063
```

Total energy

```
************************************************************
      Total energy (Hartree) at MD = 1
************************************************************

  Uele.         -3.889285101063

  Ukin.          5.533754016241
  UH0.         -14.855520072374
  UH1.           0.041395625260
  Una.          -5.040583803800
  Unl.          -0.134640939010
  Uxc0.         -1.564720823137
  Uxc1.         -1.564720823137
  Ucore.         9.551521413583
  Uhub.          0.000000000000
  Ucs.           0.000000000000
  Uzs.           0.000000000000
  Uzo.           0.000000000000
  Uef.           0.000000000000
  UvdW           0.000000000000
  Utot.         -8.033515406373
```

KS eigenvalues

```
************************************************************
************************************************************
          Eigenvalues (Hartree) for SCF KS-eq.
************************************************************
************************************************************

   Chemical Potential (Hartree) =   0.00000000000000
   Number of States             =   8.00000000000000
   HOMO =  4
   Eigenvalues
                Up-spin           Down-spin
         1  -0.69897190537228  -0.69897190537228
         2  -0.41522646150979  -0.41522646150979
         3  -0.41522645534084  -0.41522645534084
         4  -0.41521772830844  -0.41521772830844
         5   0.21218282298348   0.21218282298348
         6   0.21218282358344   0.21218282358344
         7   0.21227055734372   0.21227055734372
         8   0.24742493684297   0.24742493684297
```

Mulliken population

```
************************************************************
************************************************************
                Mulliken populations
************************************************************
************************************************************

 Total spin S =  0.000000000000

                    Up spin      Down spin      Sum          Diff
    1    C      2.509755704  2.509755704  5.019511408  0.000000000
    2    H      0.372561098  0.372561098  0.745122197  0.000000000
    3    H      0.372561019  0.372561019  0.745122038  0.000000000
    4    H      0.372561127  0.372561127  0.745122254  0.000000000
    5    H      0.372561051  0.372561051  0.745122102  0.000000000

 Sum of MulP: up   =     4.00000 down         =     4.00000
              total=     8.00000 ideal(neutral)=     8.00000
```

Standard output

```
    2    H  MulP   0.2867  0.2867 sum   0.5733
    3    H  MulP   0.2867  0.2867 sum   0.5733
    4    H  MulP   0.2867  0.2867 sum   0.5733
    5    H  MulP   0.2867  0.2867 sum   0.5733
 Sum of MulP: up   =     4.00000 down          =     4.00000
              total=     8.00000 ideal(neutral)=     8.00000
<DFT>  Total Spin Moment (muB) =  0.000000000000
<DFT>  Mixing_weight= 0.028087938921
<DFT>  Uele   =   -4.381387140168  dUele     =   0.188324995269
<DFT>  NormRD =    0.541215648239  Criterion =   0.000000000100

****************** MD= 1  SCF= 5 ******************
<Poisson>  Poisson's equation using FFT...
<Set_Hamiltonian>  Hamiltonian matrix for VNA+dVH+Vxc...
<Cluster>  Solving the eigenvalue problem...
    1    C  MulP   2.8340  2.8340 sum   5.6681
    2    H  MulP   0.2915  0.2915 sum   0.5830
    3    H  MulP   0.2915  0.2915 sum   0.5830
    4    H  MulP   0.2915  0.2915 sum   0.5830
    5    H  MulP   0.2915  0.2915 sum   0.5830
 Sum of MulP: up   =     4.00000 down          =     4.00000
              total=     8.00000 ideal(neutral)=     8.00000
<DFT>  Total Spin Moment (muB) =  0.000000000000
<DFT>  Mixing_weight= 0.028087938921
<DFT>  Uele   =   -4.352426233354  dUele     =   0.028960906814
<DFT>  NormRD =    0.509921689438  Criterion =   0.000000000100

****************** MD= 1  SCF= 6 ******************
<Poisson>  Poisson's equation using FFT...
<Set_Hamiltonian>  Hamiltonian matrix for VNA+dVH+Vxc...
<Cluster>  Solving the eigenvalue problem...
    1    C  MulP   2.5075  2.5075 sum   5.0150
    2    H  MulP   0.3731  0.3731 sum   0.7463
    3    H  MulP   0.3731  0.3731 sum   0.7463
    4    H  MulP   0.3731  0.3731 sum   0.7463
    5    H  MulP   0.3731  0.3731 sum   0.7463
 Sum of MulP: up   =     4.00000 down          =     4.00000
              total=     8.00000 ideal(neutral)=     8.00000
<DFT>  Total Spin Moment (muB) =  0.000000000000
<DFT>  Mixing_weight= 0.028087938921
<DFT>  Uele   =   -3.886371199687  dUele     =   0.466055033667
<DFT>  NormRD =    0.004026023251  Criterion =   0.000000000100

****************** MD= 1  SCF= 7 ******************
<Poisson>  Poisson's equation using FFT...
<Set_Hamiltonian>  Hamiltonian matrix for VNA+dVH+Vxc...
<Cluster>  Solving the eigenvalue problem...
```

By looking ‘met.std’ storing the standard output, one can confirm how the SCF proceeds.

Steps of MD and SCF

Mulliken population of each atom

Difference between the current and previous steps in the KS-band energy

Norm of the difference between the input and output electron densities

Convergence criterion for SCF

Calculation of binding energy of a hydrogen molecule

Calculation of a hydrogen atom

ozaki@mx17:~/openmx3.9/work

```
[ozaki@mx17 work]$ mpirun -np 1 ./openmx H.dat | tee h.std
```

You can download the input files from https://www.openmx-square.org/examples/H.dat

Calculation of a hydrogen molecule

ozaki@mx17:~/openmx3.9/work

```
[ozaki@mx17 work]$ mpirun -np 2 ./openmx H2.dat | tee h2.std
```

You can download the input files from https://www.openmx-square.org/examples/H2.dat

After finishing the two calculations, one has the following files.

ozaki@mx17:~/openmx3.9/work

```
[ozaki@mx17 work]$ ls h.*
h.cif         h.den0.cube  h.ene  h.md2  h.sden.cube  h.tden.cube  h.v1.cube     h.xyz
h.dden.cube   h.den1.cube  h.md   h.out  h.std        h.v0.cube    h.vhart.cube
[ozaki@mx17 work]$ ls h2.*
h2.Dos.val  h2.dden.cube       h2.lumo0_0_r.cube  h2.out        h2.v0.cube
h2.Dos.vec  h2.ene             h2.md              h2.std        h2.vhart.cube
h2.cif      h2.homo0_0_r.cube  h2.md2             h2.tden.cube  h2.xyz
[ozaki@mx17 work]$
```

Total energies from ‘h.out’ and ‘h2.out’

The total energies of the hydrogen atom and hydrogen molecule can be obtained from ‘h.out’ and ‘h2.out’, respectively.

Hydrogen atom	Utot.	-0.499351255995
Hydrogen molecule	Utot.	-1.165981886187

Thus, the binding energy can be calculated as

$$\text{Binding energy} = 2\,H - H_2$$
$$= 2\times(-0.49935) - (-1.16598)$$
$$= 0.1673 \text{ (Hartree)}$$
$$= 4.55 \text{ (eV)}$$

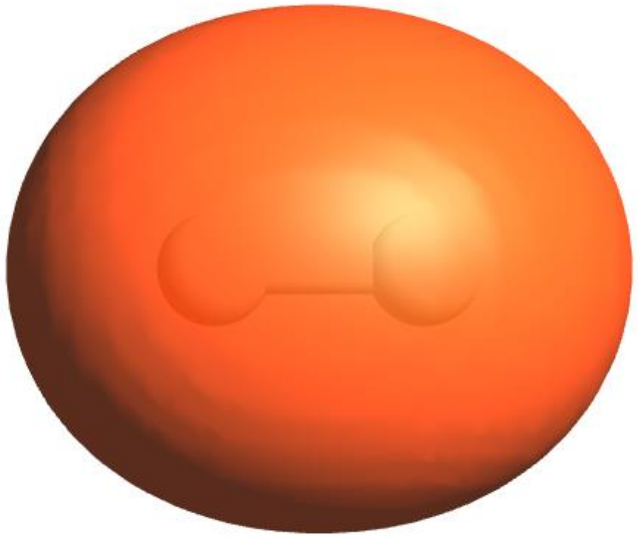
Experimental value:

4.75 (eV)

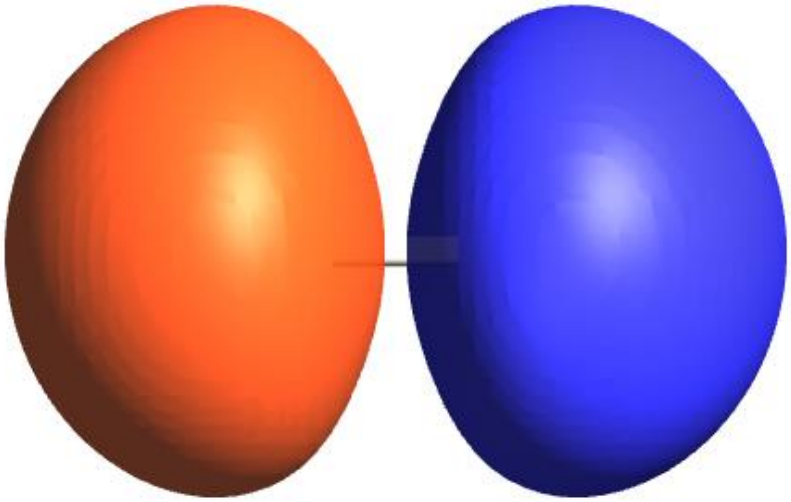
Visualization of cube files by OpenMX Viewer

After the calculation of the hydrogen molecule, the cube files for HOMO and LUMO are generated, which can be visualized by OpenMX Viewer or VESTA.

h2.homo0_0_r.cube



h2.lumo0_0_r.cube



Website of OpenMX Viewer: https://www.openmx-square.org/viewer/

Website of VESTA: https://jp-minerals.org/vesta/en

LCPAO method

(Linear-Combination of Pseudo Atomic Orbital Method)

One-particle KS orbital

$$\psi_{\sigma\mu}^{(\mathbf{k})}(\mathbf{r}) = \frac{1}{\sqrt{N}} \sum_{\mathrm{n}}^{N} \mathrm{e}^{\mathrm{i}\mathbf{R}_{\mathrm{n}}\cdot\mathbf{k}} \sum_{i\alpha} c_{\sigma\mu,i\alpha}^{(\mathbf{k})} \phi_{i\alpha}(\mathbf{r} - \tau_i - \mathbf{R}_{\mathrm{n}}),$$

is expressed by a linear combination of atomic like orbitals in the method.

$$\phi(\mathbf{r}) = Y_l^m(\hat{\mathbf{r}}) R(r)$$

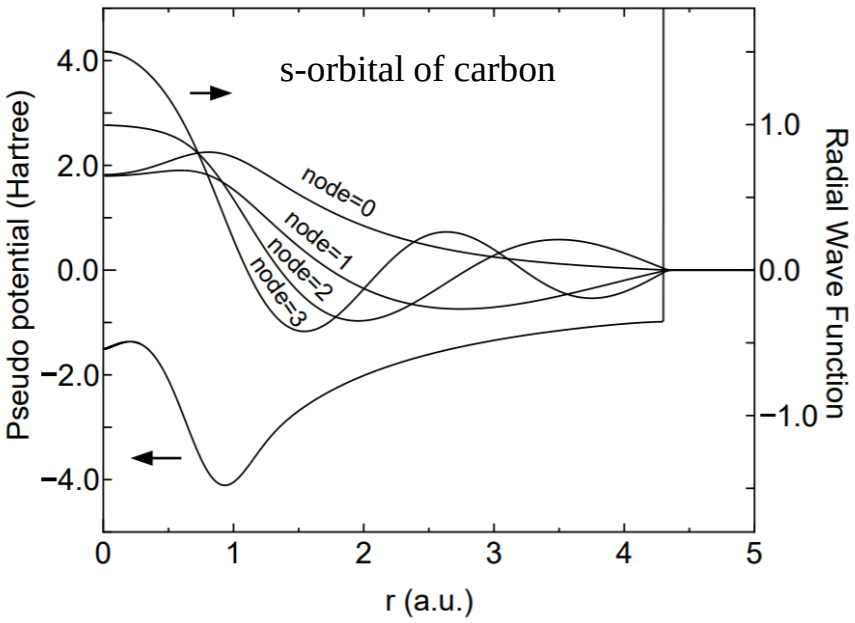
Features:

- It is easy to interpret physical and chemical meanings, since the KS orbitals are expressed by the atomic like basis functions.
- It gives rapid convergent results with respect to basis functions due to physical origin. (however, it is not a complete basis set, leading to difficulty in getting full convergence.)
- The memory and computational effort for calculation of matrix elements are O(N).
- It well matches the idea of linear scaling methods.

Primitive pseudo-atomic orbitals

1. Solve an atomic Kohn-Sham eq. under a confinement potential:

$$V_{\mathrm{core}}(r)=\begin{cases} -\dfrac{Z}{r} & \text{for } r\leqslant r_1 \\ \displaystyle\sum_{n=0}^{3} b_n r^n & \text{for } r_1<r\leqslant r_c \\ h & \text{for } r_c<r, \end{cases}$$



2. Construct the norm-conserving pseudopotentials.

3. Solve ground and excited states for the the peudopotential for each L-channel.

Ozaki, PRB 67, 155108 (2003).
Ozaki and Kino, PRB 69, 195113 (2004).

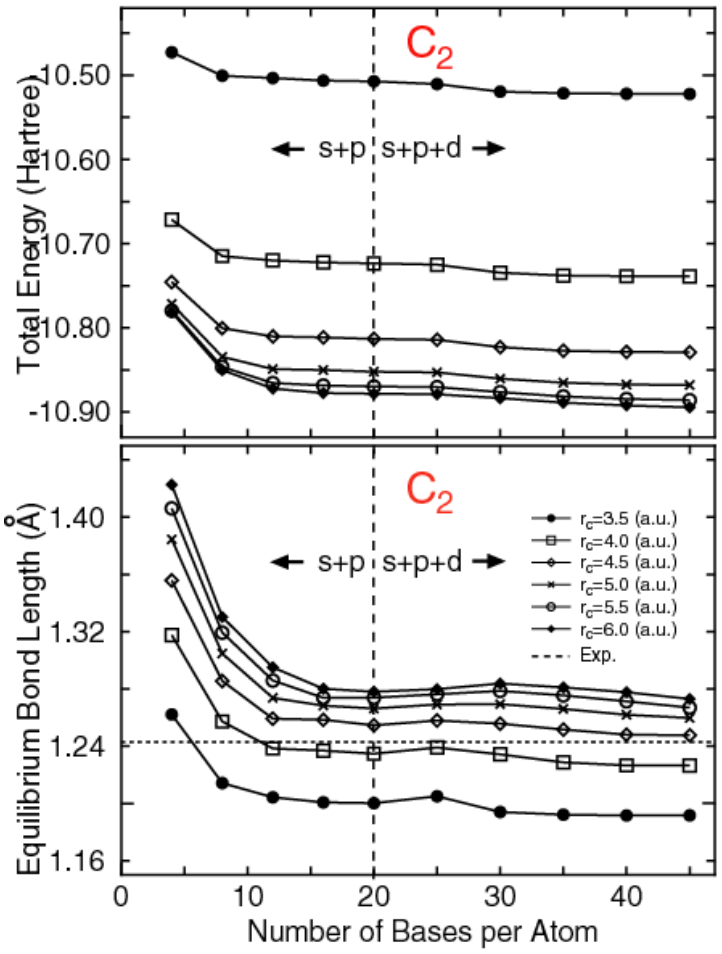
In most cases, the accuracy and efficiency can be controlled by

Cutoff radius
Number of orbitals

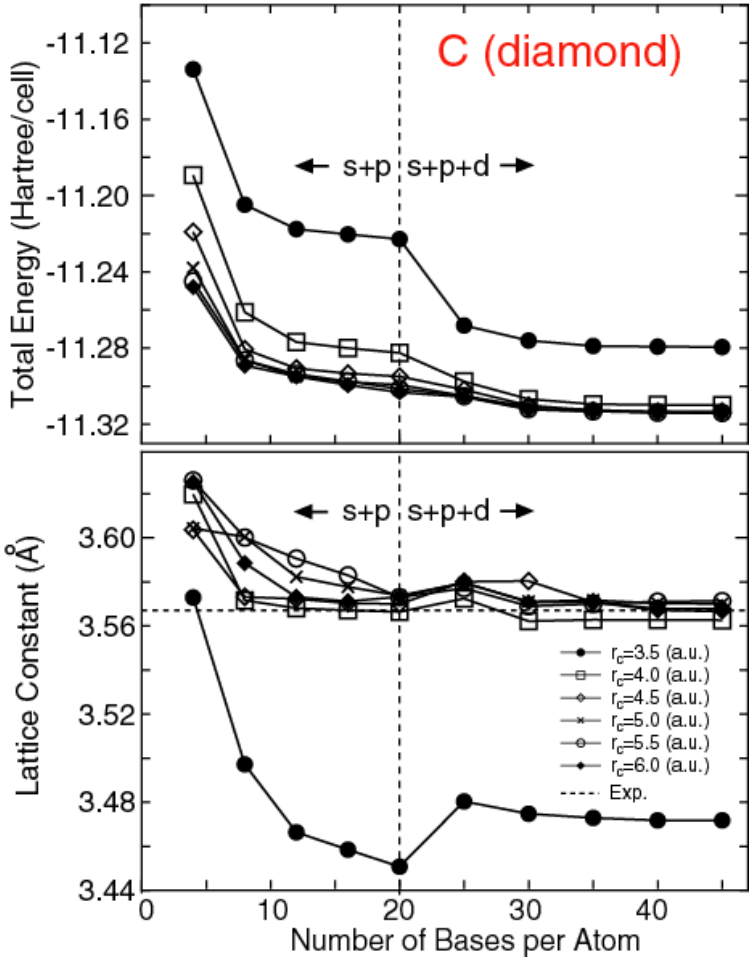
Convergence with respect to basis functions

The two parameters can be regarded as variational parameters.

Molecule



Bulk



Benchmark of primitive basis functions

Ground state calculations of dimer using primitive basis functions

Dimer	Expt.	Calc.	Dimer	Expt.	Calc.
H_2 (H4.5-$s2$)	$^1\Sigma_g^+$ [a]	$^1\Sigma_g^+$ $(1s\sigma_g^2)$	K_2 (K10.0-$s2p2$)	$^1\Sigma_g^+$ [f]	$^1\Sigma_g^+$ $(3p\pi_g^4 3p\sigma_u^2 4s\sigma_g^2)$
He_2 (He7.0-$s2$)	$^1\Sigma_g^+$ [b]	$^1\Sigma_g^+$ $(1s\sigma_g^2 1s\sigma_u^2)$	CaO (Ca7.0-$s2p2d2$)	$^1\Sigma^+$ [k]	$^1\Sigma^+$ $(s\sigma^2 s\sigma^2 p\pi^4)$
Li_2 (Li8.0-$s2$)	$^1\Sigma_g^+$ [c]	$^1\Sigma_g^+$ $(2s\sigma_g^2)$	ScO (Sc7.0-$s2p2d2$)	$^2\Sigma^+$ [l]	$^2\Sigma^+$ $(d\pi^4 s\sigma^2 s\sigma^1)$
BeO (Be6.0-$s2p2$)	$^1\Sigma^+$ [d]	$^1\Sigma^+$ $(s\sigma^2 s\sigma^2 p\pi^4)$	Ti_2 (Ti7.0-$s2p2d2$)	$^3\Delta_g$ [m]	$^3\Delta_g$ $(4s\sigma_g^2 3d\sigma_g^1 3d\pi_u^4 3d\delta_g^1)$
B_2 (B5.5-$s2p2$)	$^3\Sigma_g^-$ [e]	$^3\Sigma_g^-$ $(2s\sigma_g^2 2s\sigma_u^2 2\pi_u^2)$	V_2 (V7.5-$s2p2d2$)	$^3\Sigma_g^-$ [n]	$^1\Sigma_g^+$ $(4s\sigma_g^2 3d\sigma_g^2 3d\pi_u^4 3d\delta_g^2)$
C_2 (C5.0-$s2p2$)	$^1\Sigma_g^+$ [f]	$^1\Sigma_g^+$ $(2s\sigma_g^2 2s\sigma_u^2 2p\pi_u^4)$	V_2 (V7.5-$s4p4d4f2$)	$^3\Sigma_g^-$ [n]	$^3\Sigma_g^-$ $(4s\sigma_g^2 3d\sigma_g^2 3d\pi_u^4 3d\delta_g^2)$
N_2 (N5.0-$s2p2$)	$^1\Sigma_g^+$ [f]	$^1\Sigma_g^+$ $(2s\sigma_u^2 2p\pi_u^4 2p\sigma_g^2)$	Cr_2 (Cr7.0-$s2p2d2$)	$^1\Sigma_g^+$ [o]	$^1\Sigma_g^+$ $(4s\sigma_g^2 3d\sigma_g^2 3d\pi_u^4 3d\delta_g^4)$
O_2 (O5.0-$s2p2$)	$^3\Sigma_g^-$ [f]	$^3\Sigma_g^-$ $(2p\sigma_g^2 2p\pi_u^4 2p\pi_g^2)$	MnO (Mn7.0-$s2p2d2$)	$^6\Sigma^+$ [p]	$^6\Sigma^+$ $(d\sigma^1 d\pi^4 d\delta^2 d\pi^{*2})$
F_2 (F5.0-$s2p2$)	$^1\Sigma_g^+$ [f]	$^1\Sigma_g^+$ $(2p\sigma_g^2 2p\pi_u^4 2p\pi_g^4)$	Fe_2 (Fe7.0-$s2p2d2$)	$^7\Delta_u$ [q]	$^7\Delta_u$ $(4s\sigma_g^2 3d\sigma_g^2 3d\sigma_u^1 3d\pi_u^4 3d\pi_g^2 3d\delta_g^3 3d\delta_u^2)$
Ne_2 (Ne7.0-$s2p2$)	$^1\Sigma_g^+$ [g]	$^1\Sigma_g^+$ $(2p\pi_u^4 2p\pi_g^4 2p\sigma_u^2)$	Co_2 (Co7.0-$s2p2d2$)		$^5\Delta_g$ $(4s\sigma_g^2 3d\sigma_g^2 3d\sigma_u^1 3d\pi_u^4 3d\pi_g^2 3d\delta_g^4 3d\delta_u^3)$
Na_2 (Na9.0-$s2p2$)	$^1\Sigma_g^+$ [f]	$^1\Sigma_g^+$ $(2p\pi_g^4 2p\sigma_u^2 3s\sigma_g^2)$	Ni_2 (Ni7.0-$s2p2d2$)	Ω [r]	$^3\Sigma_g^-$ $(4s\sigma_g^2 3d\sigma_g^2 3d\sigma_u^2 3d\pi_u^4 3d\pi_g^2 3d\delta_g^4 3d\delta_u^4)$
MgO (Mg7.0-$s2p2$)	$^1\Sigma^+$ [h]	$^1\Sigma^+$ $(s\sigma^2 s\sigma^2 p\pi^4)$	Cu_2 (Cu7.0-$s2p2d2$)	$^1\Sigma_g^+$ [s]	$^1\Sigma_g^+$ $(4s\sigma_g^2 3d\sigma_g^2 3d\sigma_u^2 3d\pi_u^4 3d\pi_g^4 3d\delta_g^4 3d\delta_u^4)$
Al_2 (Al6.5-$s2p2$)	$^3\Pi_u$ [i]	$^3\Sigma_g^-$ $(3s\sigma_g^2 3s\sigma_u^2 3p\pi_u^2)$	ZnH (Zn7.0-$s2p2d2$)	$^2\Sigma_g^+$ [t]	$^2\Sigma_g^+$ $(s\sigma^2 s\sigma^{*1} d\sigma^2 d\pi^4 d\delta^4)$
Al_2 (Al6.5-$s4p4d2$)	$^3\Pi_u$ [i]	$^3\Sigma_g^-$ $(3s\sigma_g^2 3s\sigma_u^2 3p\pi_u^2)$	GaH (Ga7.0-$s2p2$)	$^1\Sigma^+$ [u]	$^1\Sigma^+$ $(s\sigma^2 s\sigma^{*2})$
Si_2 (Si6.5-$s2p2$)	$^3\Sigma_g^-$ [f]	$^3\Pi_u$ $(3s\sigma_u^2 3s\sigma_g^1 3p\pi_u^3)$	GeO (Ge7.0-$s2p2$)	$^1\Sigma^+$ [f]	$^1\Sigma^+$ $(ss\sigma^2 sp\sigma^2 pp\pi^4 pp\sigma^2)$
Si_2 (Si6.5-$s2p2d1$)	$^3\Sigma_g^-$ [f]	$^3\Sigma_g^-$ $(3s\sigma_u^2 3p\pi_u^2 3s\sigma_g^2)$	As_2 (As7.0-$s2p2d1$)	$^1\Sigma_g^+$ [f]	$^1\Sigma_g^+$ $(4s\sigma_g^2 4s\sigma_u^2 4p\sigma_g^2 4p\pi_u^4)$
P_2 (P6.0-$s2p2d1$)	$^1\Sigma_g^+$ [f]	$^1\Sigma_g^+$ $(3s\sigma_u^2 3p\sigma_g^2 3p\pi_u^4)$	Se_2 (Se7.0-$s2p2d1$)	$^3\Sigma_g^-$ [f]	$^3\Sigma_g^-$ $(4s\sigma_g^2 4s\sigma_u^2 4p\sigma_g^2 4p\pi_u^4 4p\pi_g^2)$
S_2 (S6.0-$s2p2$)	$^3\Sigma_g^-$ [f]	$^3\Sigma_g^-$ $(3p\sigma_g^2 3p\pi_u^4 3p\pi_g^2)$	Br_2 (Br7.0-$s2p2d1$)	$^1\Sigma_g^+$ [f]	$^1\Sigma_g^+$ $(4s\sigma_g^2 4s\sigma_u^2 4p\sigma_g^2 4p\pi_u^4 4p\pi_g^4)$
Cl_2 (Cl6.0-$s2p2d2$)	$^1\Sigma_g^+$ [f]	$^1\Sigma_g^+$ $(3p\sigma_g^2 3p\pi_u^4 3p\pi_g^4)$	Kr_2 (Kr7.0-$s2p2$)	$^1\Sigma_g^+$ [v]	$^1\Sigma_g^+$ $(4s\sigma_g^2 4s\sigma_u^2 4p\sigma_g^2 4p\sigma_u^2 4p\pi_u^4 4p\pi_g^4)$
Ar_2 (Ar7.0-$s2p2$)	$^1\Sigma_g^+$ [j]	$^1\Sigma_g^+$ $(3p\pi_u^4 3p\pi_g^4 3p\sigma_u^2)$			

Ozaki and Kino, PRB 69, 195113 (2004).

All the successes and failures by the LDA are reproduced by the modest size of basis functions (DNP in most cases).

Variational optimization of basis functions

One-particle wave functions

$$\psi_\mu(\mathbf{r}) = \sum_{i\alpha} c_{\mu,i\alpha}\phi_{i\alpha}(\mathbf{r}-\mathbf{r}_i)$$

Contracted orbitals

$$\phi_{i\alpha}(\mathbf{r}) = \sum_q a_{i\alpha q}\chi_{i\eta}(\mathbf{r})$$

The variation of E with respect to c with fixed a gives

$$\partial E_{\mathrm{tot}} / \partial c_{\mu,i\alpha} = 0 \quad \rightarrow \quad \sum_{j\beta}\langle\phi_{i\alpha}|\hat{H}|\phi_{j\beta}\rangle c_{\mu,j\beta} = \varepsilon_\mu \sum_{j\beta}\langle\phi_{i\alpha}|\phi_{j\beta}\rangle c_{\mu,j\beta}$$

Regarding c as dependent variables on a and assuming KS eq. is solved self-consistently with respect to c, we have

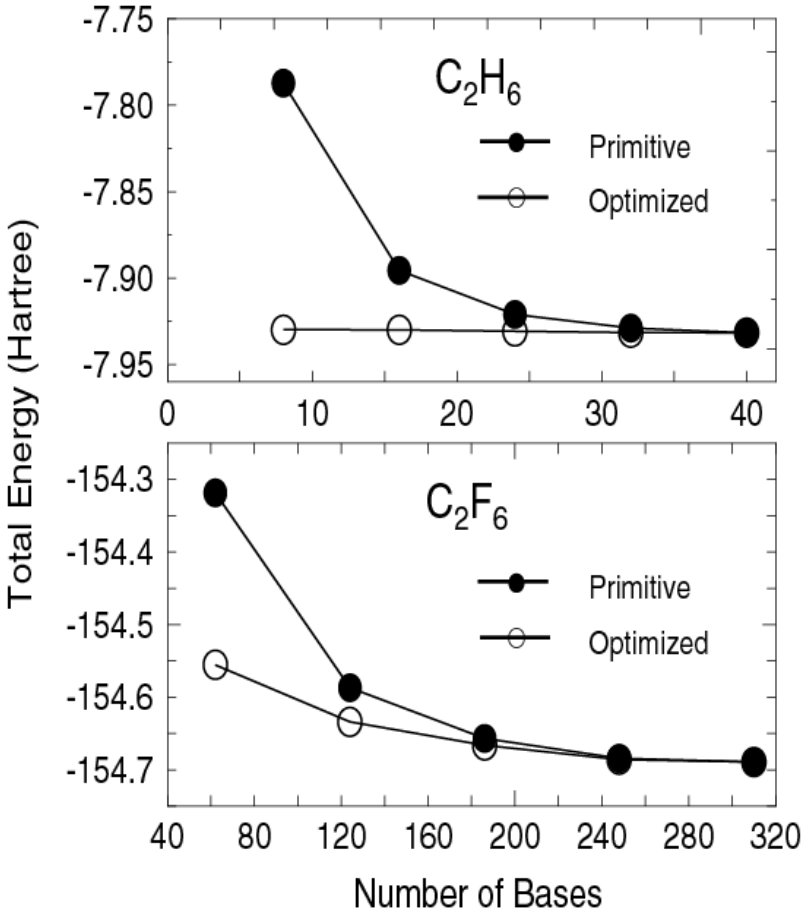
$$\frac{\partial E_{\mathrm{tot}}}{\partial a_{i\alpha q}} = \int dr^3 \frac{\delta E_{\mathrm{tot}}}{\delta\rho(\mathbf{r})}\frac{\delta\rho(\mathbf{r})}{\delta a_{i\alpha q}}$$

$$= 2\left(\sum_{j\beta}\Theta_{i\alpha,j\beta}\langle\chi_{i\eta}|\hat{H}|\phi_{j\beta}\rangle - E_{i\alpha,j\beta}\langle\chi_{i\eta}|\phi_{j\beta}\rangle\right)$$

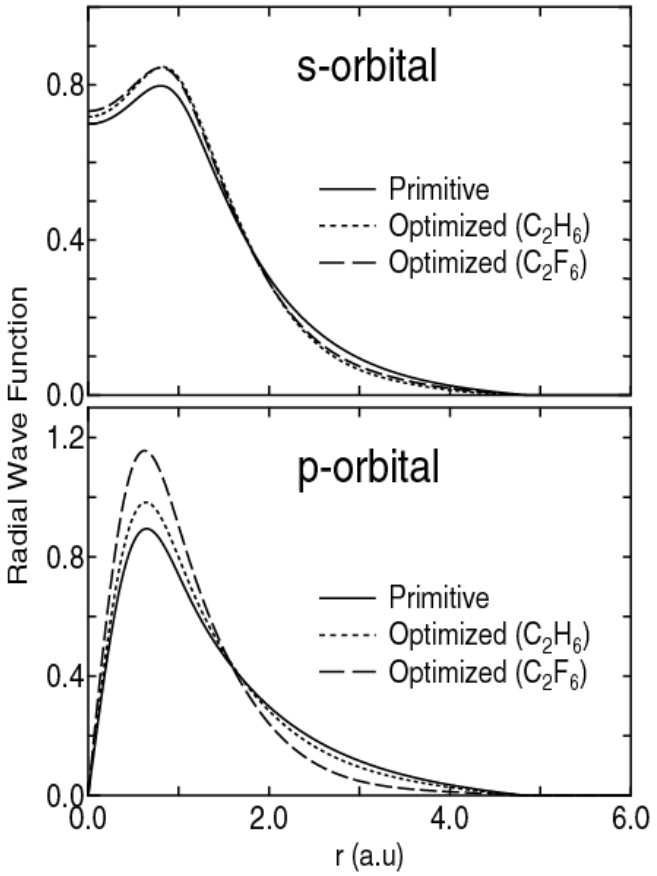
Ozaki, PRB 67, 155108 (2003).

Primitive vs. Optimized

Energy convergence



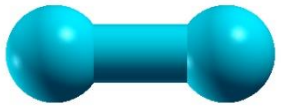
Radial shape in carbon atom



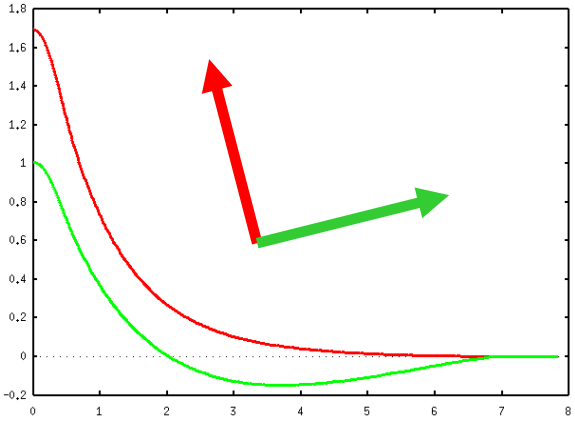
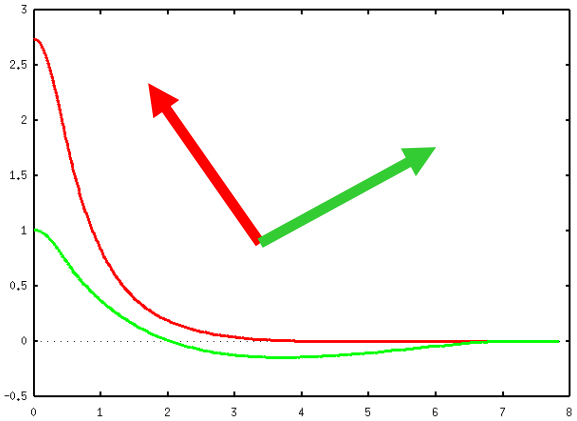
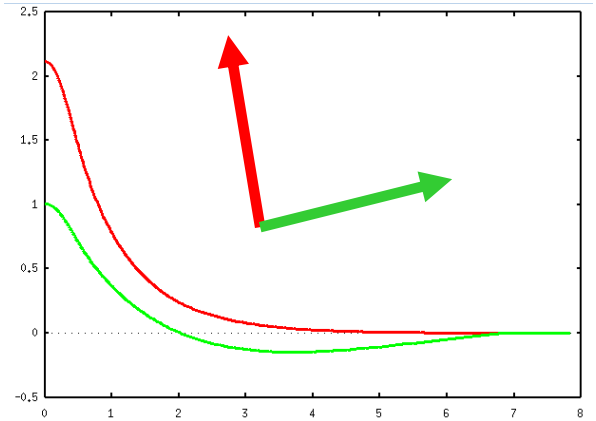
Fluorine atoms pull electrons out of the p orbitals of carbon atoms. So, the p orbital of the carbon atom is largely localized.

Optimization of basis functions

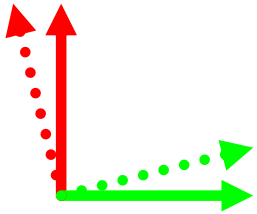
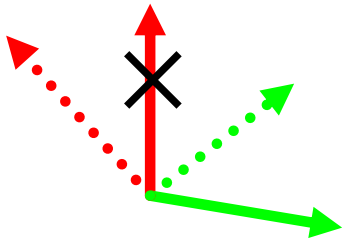
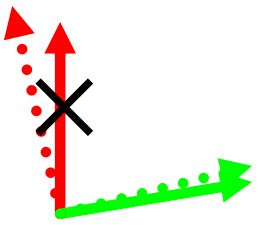
1. Choose typical chemical environments


2. Optimize variationally the radial functions

3. Rotate a set of optimized orbitals within the subspace, and discard the redundant funtions

Database of optimized VPS and PAO

Database (2019) of optimized VPS and PAO

The database (2019) of fully relativistic pseudopotentials (VPS) and pseudo-atomic orbitals (PAO), generated by ADPACK, which could be an input data of program package, OpenMX. The data of elements with the underline are currently available. When you use these data, VPS and PAO, in the program package, OpenMX, then copy them to the directory, openmx*.*/DFT_DATA19/VPS/ and openmx*.*/DFT_DATA19/PAO/, respectively. The delta gauge of OpenMX with the database (2019) is found at here.

E **Guideline for choice of PAOs**

H **See also the pages 51-60 in the tutorial material** He

Li	Be											B	C	N	O	F	Ne
Na	Mg											Al	Si	P	S	Cl	Ar
K	Ca	Sc	Ti	V	Cr	Mn	Fe	Co	Ni	Cu	Zn	Ga	Ge	As	Se	Br	Kr
Rb	Sr	Y	Zr	Nb	Mo	Tc	Ru	Rh	Pd	Ag	Cd	In	Sn	Sb	Te	I	Xe
Cs	Ba	L	Hf	Ta	W	Re	Os	Ir	Pt	Au	Hg	Tl	Pb	Bi	Po	At	Rn
Fr	Ra	A															

L	La	Ce	Pr	Nd	Pm	Sm	Eu	Gd	Tb	Dy	Ho	Er	Tm	Yb	Lu
A	Ac	Th	Pa	U	Np	Pu	Am	Cm	Bk	Cf	Es	Fm	Md	No	Lr

http://www.openmx-square.org/

Implementation: Total energy (1)

The total energy is given by the sum of six terms, and a proper integration scheme for each term is applied to accurately evaluate the total energy.

$$E_{\rm tot} = E_{\rm kin} + E_{\rm ec} + E_{\rm ee} + E_{\rm xc} + E_{\rm cc} = E_{\rm kin} + E_{\rm na} + E_{\rm ec}^{\rm (NL)} + E_{\delta\rm ee} + E_{\rm xc} + E_{\rm scc}.$$

$$E_{\rm kin} = \sum_{\sigma}\sum_{\rm n}^{N}\sum_{i\alpha,j\beta}\rho_{\sigma,i\alpha j\beta}^{(\mathbf{R}_{\rm n})}h_{i\alpha j\beta,{\rm kin}}^{(\mathbf{R}_{\rm n})}.$$ Kinetic energy

$$E_{\rm ec} = E_{\rm ec}^{\rm (L)} + E_{\rm ec}^{\rm (NL)},$$ Coulomb energy with external potential

$$= \sum_{\sigma}\sum_{\rm n}^{N}\sum_{i\alpha,j\beta}\rho_{\sigma,i\alpha j\beta}^{(\mathbf{R}_{\rm n})}\langle\phi_{i\alpha}(\mathbf{r}-\tau_i)|\sum_I V_{{\rm core},I}(\mathbf{r}-\tau_I)|\phi_{j\beta}(\mathbf{r}-\tau_j-\mathbf{R}_{\rm n})\rangle$$

$$+ \sum_{\sigma}\sum_{\rm n}^{N}\sum_{i\alpha,j\beta}\rho_{\sigma,i\alpha j\beta}^{(\mathbf{R}_{\rm n})}\langle\phi_{i\alpha}(\mathbf{r}-\tau_i)|\sum_I V_{{\rm NL},I}(\mathbf{r}-\tau_I)|\phi_{j\beta}(\mathbf{r}-\tau_j-\mathbf{R}_{\rm n})\rangle,$$

$$E_{\rm ee} = \frac{1}{2}\int dr^3 n(\mathbf{r})V_{\rm H}(\mathbf{r}),$$ Hartree energy

$$= \frac{1}{2}\int dr^3 n(\mathbf{r})\{V_{\rm H}^{\rm (a)}(\mathbf{r}) + \delta V_{\rm H}(\mathbf{r})\},$$

$$E_{\rm xc} = \int dr^3\{n_\uparrow(\mathbf{r}) + n_\downarrow(\mathbf{r}) + n_{\rm pcc}(\mathbf{r})\}\epsilon_{\rm xc}(n_\uparrow + \frac{1}{2}n_{\rm pcc}, n_\downarrow + \frac{1}{2}n_{\rm pcc}),$$ Exchange-correlation energy

$$E_{\rm cc} = \frac{1}{2}\sum_{I,J}\frac{Z_I Z_J}{|\tau_I - \tau_J|}.$$ Core-core Coulomb energy

TO and H.Kino, PRB 72, 045121 (2005)

Implementation: Total energy (2)

The reorganization of Coulomb energies gives three new energy terms.

$$E_{\rm ec}^{(L)} + E_{\rm ee} + E_{\rm cc} = E_{\rm na} + E_{\delta\rm ee} + E_{\rm scc},$$

The neutral atom energy

$$\begin{aligned} E_{\rm na} &= \int dr^3 n(\mathbf{r}) \sum_I V_{{\rm na},I}(\mathbf{r}-\tau_I), \\ &= \sum_\sigma \sum_{\rm n}^{N} \sum_{i\alpha,j\beta} \rho_{\sigma,i\alpha j\beta}^{(\mathbf{R_n})} \sum_I \langle \phi_{i\alpha}(\mathbf{r}-\tau_i) | V_{{\rm na},I}(\mathbf{r}-\tau_I) | \phi_{j\beta}(\mathbf{r}-\tau_j-\mathbf{R}_{\rm n}) \rangle, \end{aligned}$$

Short range and separable to two-center integrals

Difference charge Hartree energy

$$E_{\delta\rm ee} = \frac{1}{2}\int dr^3 \delta n(\mathbf{r}) \delta V_{\rm H}(\mathbf{r}),$$

Long range but minor contribution

Screened core-core repulsion energy

$$E_{\rm scc} = \frac{1}{2}\sum_{I,J}\left[\frac{Z_I Z_J}{|\tau_I-\tau_J|} - \int dr^3 n_I^{(\rm a)}(\mathbf{r}) V_{{\rm H},J}^{(\rm a)}(\mathbf{r})\right].$$

Short range and two-center integrals

Difference charge

$$\begin{aligned} \delta n(\mathbf{r}) &= n(\mathbf{r}) - n^{(\rm a)}(\mathbf{r}), \\ &= n(\mathbf{r}) - \sum_i n_i^{(\rm a)}(\mathbf{r}), \end{aligned}$$

Neutral atom potential

$$V_{{\rm na},I}(\mathbf{r}-\tau_I) = V_{{\rm core},I}(\mathbf{r}-\tau_I) + V_{{\rm H},I}^{(\rm a)}(\mathbf{r}-\tau_I).$$

Implementation: Total energy (3)

So, the total energy is given by

$$E_{\mathrm{tot}} = E_{\mathrm{kin}} + E_{\mathrm{na}} + E_{\mathrm{ec}}^{(\mathrm{NL})} + E_{\delta\mathrm{ee}} + E_{\mathrm{xc}} + E_{\mathrm{scc}}.$$

Each term is evaluated by using a different numerical grid with consideration on accuracy and efficiency.

E_{kin}
E_{na} } Spherical coordinate in momentum space
$E_{\mathrm{ec}}^{(\mathrm{NL})}$

$E_{\delta\mathrm{ee}}$
E_{xc} } Real space regular mesh

E_{scc} Real space fine mesh

Two center integrals

Fourier-transformation of basis functions

$$
\begin{aligned}
\tilde{\phi}_{i\alpha}(\mathbf{k}) &= \left(\frac{1}{\sqrt{2\pi}}\right)^3 \int dr^3 \phi_{i\alpha}(\mathbf{r}) \mathrm{e}^{-\mathrm{i}\mathbf{k}\cdot\mathbf{r}} \\
&= \left(\frac{1}{\sqrt{2\pi}}\right)^3 \int dr^3 Y_{lm}(\hat{\mathbf{r}}) R_{pl}(r) \left\{ 4\pi \sum_{L=0}^{\infty} \sum_{M=-L}^{L} (-\mathrm{i})^L j_L(kr) Y_{LM}(\hat{\mathbf{k}}) Y^*_{LM}(\hat{\mathbf{r}}) \right\}, \\
&= \left(\frac{1}{\sqrt{2\pi}}\right)^3 4\pi \sum_{L=0}^{\infty} \sum_{M=-L}^{L} (-\mathrm{i})^L Y_{LM}(\hat{\mathbf{k}}) \int dr r^2 R_{pl}(r) j_L(kr) \int d\theta d\phi \sin(\theta) Y_{lm}(\hat{\mathbf{r}}) Y^*_{LM}(\hat{\mathbf{r}}), \\
&= \left[\left(\frac{1}{\sqrt{2\pi}}\right)^3 4\pi (-\mathrm{i})^l \int dr r^2 R_{pl}(r) j_L(kr) \right] Y_{lm}(\hat{\mathbf{k}}), \\
&= \tilde{R}_{pl}(k) Y_{lm}(\hat{\mathbf{k}}),
\end{aligned}
$$

Integrals for angular parts are analytically performed. Thus, we only have to perform one-dimensional integrals along the radial direction.

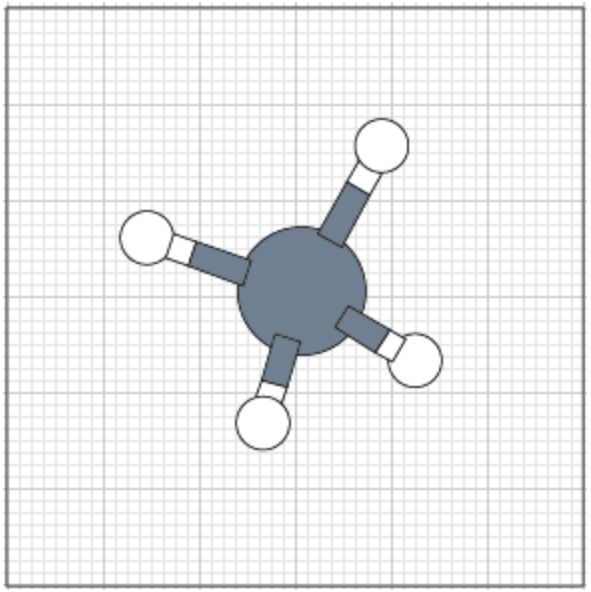
e.g., overlap integral

$$
\begin{aligned}
\langle \phi_{i\alpha}(\mathbf{r}) | \phi_{j\beta}(\mathbf{r}-\tau) \rangle &= \int dr^3 \phi^*_{i\alpha}(\mathbf{r}) \phi_{j\beta}(\mathbf{r}-\tau), \\
&= \int dr^3 \left(\frac{1}{\sqrt{2\pi}}\right)^3 \int dk^3 \tilde{R}^*_{pl}(k) Y^*_{lm}(\hat{\mathbf{k}}) \mathrm{e}^{-\mathrm{i}\mathbf{k}\cdot\mathbf{r}} \left(\frac{1}{\sqrt{2\pi}}\right)^3 \int dk'^3 \tilde{R}_{p'l'}(k') Y_{l'm'}(\hat{\mathbf{k}}') \mathrm{e}^{\mathrm{i}\mathbf{k}'\cdot(\mathbf{r}-\tau)}, \\
&= \left(\frac{1}{2\pi}\right)^3 \int dk^3 \int dk'^3 \mathrm{e}^{-\mathrm{i}\mathbf{k}'\cdot\tau} \tilde{R}^*_{pl}(k) Y^*_{lm}(\hat{\mathbf{k}}) \tilde{R}_{p'l'}(k') Y_{l'm'}(\hat{\mathbf{k}}') \int dr^3 \mathrm{e}^{\mathrm{i}(\mathbf{k}'-\mathbf{k})\cdot\mathbf{r}}, \\
&= \int dk^3 \mathrm{e}^{-\mathrm{i}\mathbf{k}\cdot\tau} \tilde{R}^*_{pl}(k) Y^*_{lm}(\hat{\mathbf{k}}) \tilde{R}_{p'l'}(k) Y_{l'm'}(\hat{\mathbf{k}}),
\end{aligned}
$$

Cutoff energy for regular mesh

The two energy components $E_{\delta ee} + E_{xc}$ are calculated on real space regular mesh. The mesh fineness is determined by plane-wave cutoff energies.

```
scf.energycutoff          150.0          # default=150 (Ry)
```



The cutoff energy can be related to the mesh fineness by the following eqs.

$$E_{\rm cut}^{(1)} = \frac{1}{2}\mathbf{gb}_1 \cdot \mathbf{gb}_1, \quad E_{\rm cut}^{(2)} = \frac{1}{2}\mathbf{gb}_2 \cdot \mathbf{gb}_2, \quad E_{\rm cut}^{(3)} = \frac{1}{2}\mathbf{gb}_3 \cdot \mathbf{gb}_3,$$

$$\mathbf{ga}_1 = \frac{\mathbf{a}_1}{N_1}, \quad \mathbf{ga}_2 = \frac{\mathbf{a}_2}{N_2}, \quad \mathbf{ga}_3 = \frac{\mathbf{a}_3}{N_3},$$

$$\mathbf{gb}_1 = 2\pi\frac{\mathbf{ga}_2 \times \mathbf{ga}_2}{\Delta V}, \quad \mathbf{gb}_2 = 2\pi\frac{\mathbf{ga}_3 \times \mathbf{ga}_1}{\Delta V}, \quad \mathbf{gb}_2 = 2\pi\frac{\mathbf{ga}_1 \times \mathbf{ga}_2}{\Delta V},$$

$$\Delta V = \mathbf{ga}_1 \cdot (\mathbf{ga}_2 \times \mathbf{ga}_3),$$

Forces

$$\mathbf{F}_i = -\frac{\partial E_{\rm tot}}{\partial \mathbf{R}_i} = -\frac{\partial E_{\rm kin}}{\partial \mathbf{R}_i} - \frac{\partial E_{\rm na}}{\partial \mathbf{R}_i} - \frac{\partial E_{\rm scc}}{\partial \mathbf{R}_i} - \frac{\partial E_{\delta \rm ee}}{\partial \mathbf{R}_i} - \frac{\partial E_{\rm xc}}{\partial \mathbf{R}_i} - \frac{\partial E_{\rm cc}}{\partial \mathbf{R}_i}$$

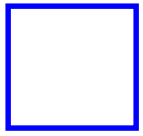
$$\frac{\partial E_{\delta \rm ee}}{\partial \mathbf{R}_k} = \sum_{\rm p} \frac{\partial n(\mathbf{r}_{\rm p})}{\partial \mathbf{R}_k} \frac{\partial E_{\delta \rm ee}}{\partial n(\mathbf{r}_{\rm p})} + \sum_{\rm p} \frac{\partial n^{\rm a}(\mathbf{r}_{\rm p})}{\partial \mathbf{R}_k} \frac{\partial E_{\delta \rm ee}}{\partial n^{\rm a}(\mathbf{r}_{\rm p})}.$$

$$\begin{aligned}
\frac{\partial E_{\delta \rm ee}}{\partial n(\mathbf{r}_{\rm p})} &= \frac{1}{2}\Delta V \{\delta V_{\rm H}(\mathbf{r}_{\rm p}) + \sum_{\rm q} \delta n(\mathbf{r}_{\rm q}) \frac{\partial \delta V_{\rm H}(\mathbf{r}_{\rm q})}{\partial n(\mathbf{r}_{\rm p})}\}, \\
&= \frac{1}{2}\Delta V \{\delta V_{\rm H}(\mathbf{r}_{\rm p}) + \frac{4\pi}{N_{\rm rsg}} \sum_{\rm G} \frac{1}{|\mathbf{G}|^2} \sum_{\rm q} \delta n(\mathbf{r}_{\rm q}) e^{i\mathbf{G}\cdot(\mathbf{r}_{\rm q}-\mathbf{r}_{\rm p})}\}, \\
&= \Delta V \delta V_{\rm H}(\mathbf{r}_{\rm p}).
\end{aligned}$$

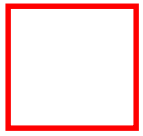
$$\begin{aligned}
\frac{\partial E_{\delta \rm ee}}{\partial n^{\rm a}(\mathbf{r}_{\rm p})} &= -\frac{1}{2}\Delta V \{\delta V_{\rm H}(\mathbf{r}_{\rm p}) - \sum_{\rm q} \delta n(\mathbf{r}_{\rm q}) \frac{\partial \delta V_{\rm H}(\mathbf{r}_{\rm q})}{\partial n^{\rm a}(\mathbf{r}_{\rm p})}\}, \\
&= -\frac{1}{2}\Delta V \{\delta V_{\rm H}(\mathbf{r}_{\rm p}) + \frac{4\pi}{N_{\rm rsg}} \sum_{\rm G} \frac{1}{|\mathbf{G}|^2} \sum_{\rm q} \delta n(\mathbf{r}_{\rm q}) e^{i\mathbf{G}\cdot(\mathbf{r}_{\rm q}-\mathbf{r}_{\rm p})}\}, \\
&= -\Delta V \delta V_{\rm H}(\mathbf{r}_{\rm p}).
\end{aligned}$$

$$\begin{aligned}
\frac{\partial n(\mathbf{r}_{\rm p})}{\partial \mathbf{R}_k} &= \sum_{i\alpha,j\beta} \sum_{\nu} \{\frac{\partial c^*_{i\alpha,\nu}}{\partial \mathbf{R}_k} c_{j\beta,\nu} \chi_{i\alpha}(\mathbf{r}) \chi_{j\beta}(\mathbf{r}) + c^*_{i\alpha,\nu} \frac{\partial c_{j\beta,\nu}}{\partial \mathbf{R}_k} \chi_{i\alpha}(\mathbf{r}_{\rm p}) \chi_{j\beta}(\mathbf{r}_{\rm p})\} \\
&\quad + 2 \sum_{\alpha,j\beta} \rho_{k\alpha,j\beta} \frac{\partial \chi_{k\alpha}(\mathbf{r}_{\rm p})}{\partial \mathbf{R}_k} \chi_{j\beta}(\mathbf{r}_{\rm p}).
\end{aligned}$$

$$\begin{aligned}
\frac{\partial E_{\rm xc}}{\partial \mathbf{R}_k} &= \sum_{\rm p} \frac{\partial n(\mathbf{r}_{\rm p})}{\partial \mathbf{R}_k} \frac{\partial E_{\rm xc}}{\partial n(\mathbf{r}_{\rm p})}, \\
&= \Delta V \sum_{\rm p} \frac{\partial n(\mathbf{r}_{\rm p})}{\partial \mathbf{R}_k} v_{\rm xc}(n(\mathbf{r}_{\rm p})).
\end{aligned}$$



Easy calc.



See the left

Forces are always analytic at any grid fineness and at zero temperature, even if numerical basis functions and numerical grids.

Norm-conserving pseudopotential by MBK

I. Morrion, D.M. Bylander, and L. Kleinman, PRB 47, 6728 (1993).

If $Q_{ij} = 0$, the non-local terms can be transformed to a diagonal form.

$$V_{\rm NL} = \sum_{i,j} B_{ij} |\beta_i\rangle\langle\beta_j|,$$
$$= \sum_{i} \lambda_i |\alpha_i\rangle\langle\alpha_i|$$

The form is equivalent to that obtained from the Blochl expansion for TM norm-conserving pseudopotentials. Thus, common routines can be utilized for the MBK and TM pseudopotentials, resulting in easiness of the code development.

To satisfy $Q_{ij}=0$, pseudofunctions are now given by

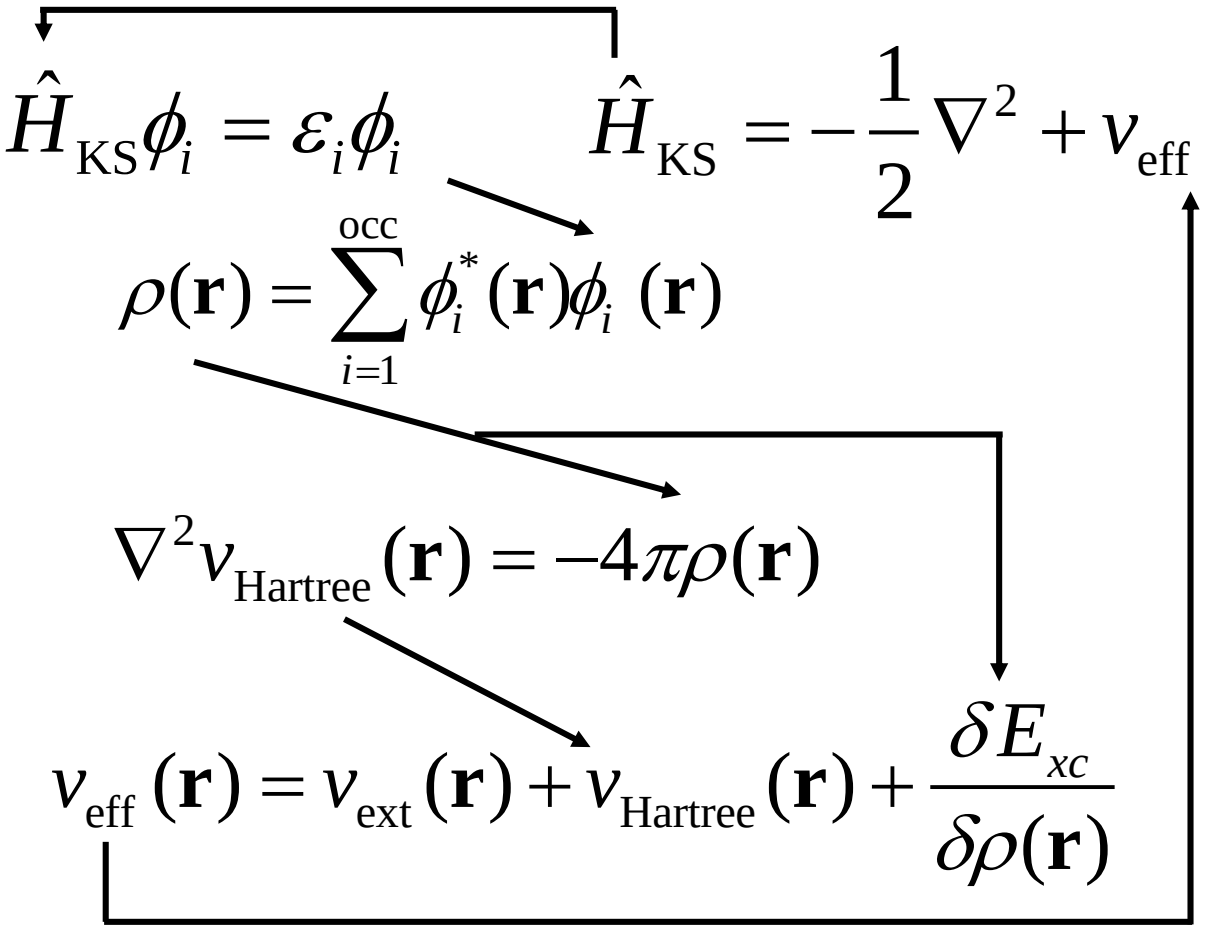
$$\phi_i = \phi_{{\rm TM},i} + f_i \quad f_i = \sum_{i=0} c_i \left[r j_l \left(\frac{r}{r_c} u_{li} \right) \right]$$

The coefficients {c} are determined by agreement of derivatives and $Q_{ij}=0$.
Once a set of {c} is determined, χ is given by

$$\chi_i = V_{\rm TM}^{(i)} \phi_{{\rm TM},i} + \varepsilon_i f_i - V_{\rm loc}\phi_i - \frac{1}{2}\sum_i c_i \left(\frac{u_{li}}{r_c} \right)^2 \left[r j_l \left(\frac{r}{r_c} u_{li} \right) \right]$$

Algorithmic structure of KS eq.

3D coupled non-linear differential equations have to be solved self-consistently.



Input charge = Output charge → Self-consistent condition

Self-consistency: Simple charge mixing

The KS effective potential is constructed from ρ.
However, ρ is evaluated from eigenfunctions of KS eq.

$$\hat{H}_{\mathrm{KS}}\phi_i = \varepsilon_i\phi_i \qquad \hat{H}_{\mathrm{KS}} = -\frac{1}{2}\nabla^2 + v_{\mathrm{eff}}$$

$$v_{\mathrm{eff}} = v_{\mathrm{ext}}(\mathbf{r}) + v_{\mathrm{Hartree}}(\mathbf{r}) + \frac{\delta E_{\mathrm{xc}}}{\delta\rho(\mathbf{r})}$$

$$\rho(\mathbf{r}) = \sum_i \phi_i^*(\mathbf{r})\phi_i(\mathbf{r})$$

Simple charge mixing method

The next input density is constructed by a simple mixing of input and output densities.

$$\rho_{n+1}^{(\mathrm{in})} = \alpha\rho_n^{(\mathrm{in})} + (1-\alpha)\rho_n^{(\mathrm{out})},$$

It works well for large gap systems and small sized systems.

Self-consistency: RMM-DIIS

Idea:

Minimize the norm of a linear combination of previous residual vectors.

$$\bar{R}_{n+1} = \sum_{m=n-(p-1)}^{n} \alpha_m R_m,$$

$$\begin{aligned} F &= \langle \bar{R}_{n+1} | \bar{R}_{n+1} \rangle - \lambda \left(1 - \sum_m^n a_m \right), \\ &= \sum_{m,m'} \alpha_m \alpha_{m'} \langle R_m | R_{m'} \rangle - \lambda \left(1 - \sum_m^n a_m \right). \end{aligned}$$

$$R_n(\mathbf{q}) \equiv \tilde{n}_n^{(\mathrm{out})}(\mathbf{q}) - \tilde{n}_n^{(\mathrm{in})}(\mathbf{q}),$$

$$\langle R_m | R_{m'} \rangle \equiv \sum_{\mathbf{q}} \frac{R_m^*(\mathbf{q}) R_{m'}(\mathbf{q})}{w(\mathbf{q})},$$

Kerker factor

$$w(\mathbf{q}) = \frac{|\mathbf{q}|^2}{|\mathbf{q}|^2 + q_0^2},$$

Long wave length components corresponding to small |q| are taken into account.

Minimization of F leads to

$$\frac{\partial F}{\partial \alpha_k} = 0 \qquad \frac{\partial F}{\partial \lambda} = 0$$

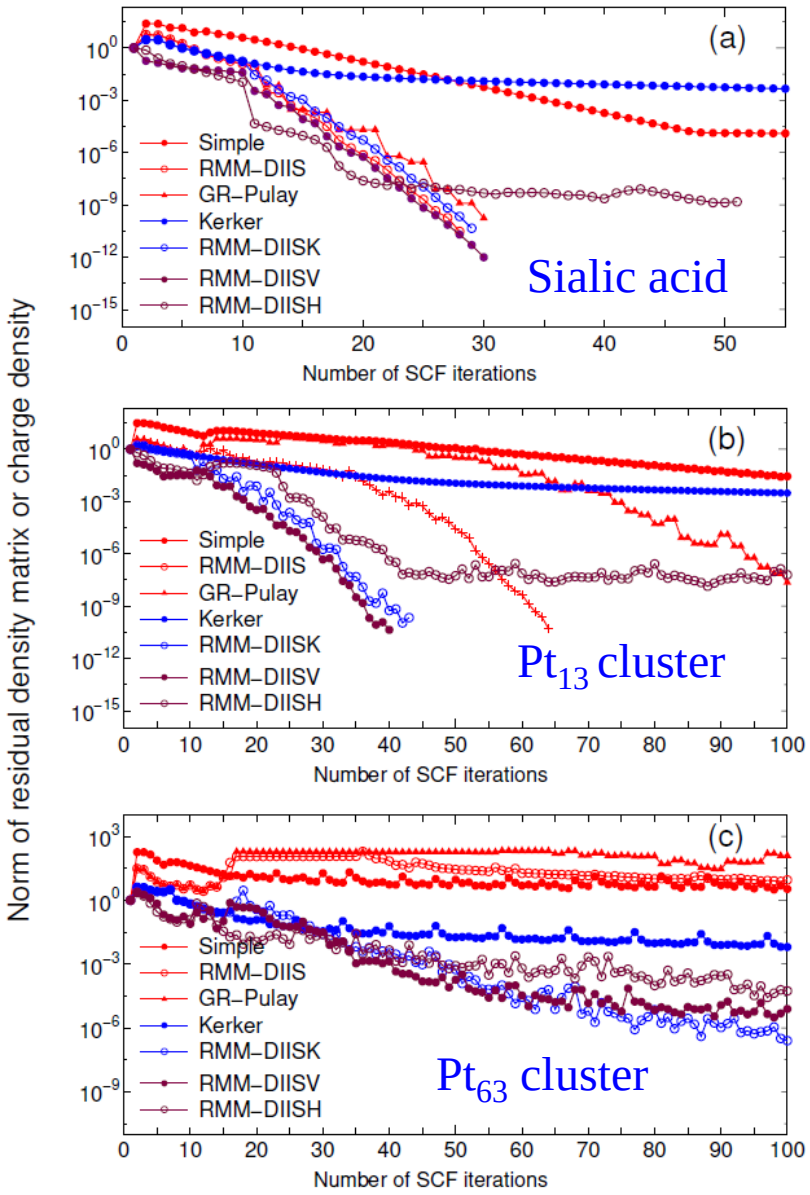
$$\begin{pmatrix} \langle R_{n-(p-1)} | R_{n-(p-1)} \rangle & \cdots & \cdots & 1 \\ \cdots & \cdots & \cdots & 1 \\ \cdots & \cdots & \langle R_n | R_n \rangle & \cdots \\ 1 & 1 & \cdots & 0 \end{pmatrix} \begin{pmatrix} \alpha_{n-(p-1)} \\ \alpha_{n-(p-1)+1} \\ \cdot \\ \frac{1}{2}\lambda \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \cdot \\ 1 \end{pmatrix}.$$

Optimum input density might be given by

$$\rho_{n+1}^{(\mathrm{in})} = \sum_{m=n-(p-1)}^{n} \alpha_m \rho_m^{(\mathrm{in})} + \beta \sum_{m=n-(p-1)}^{n} \alpha_m R_m$$

G.Kresse and J. Furthmeuller, PRB 54, 11169 (1996).

Mixing methods



Available mixing methods:

Simple mixing (Simple)

Residual minimization method in the direct inversion iterative subspace (RMM-DIIS)

Guaranteed reduction Pulay method (GR-Pulay)

Kerker mixing (Kerker)

RMM-DIIS with Kerker metric (RMM-DIISK)

RMM-DIIS for Kohn-Sham potential (RMM-DIISV)

RMM-DIIS for Hamiltonian (RMM-DIISH)

Recommendation:
RMM-DIISK or RMM-DIISV

For DFT+U and constrained methods
RMM-DIISH

See also the manual at

https://www.openmx-square.org/openmx_man3.9/node40.html

Specification of PAO and VPS

PAO and VPS are specified by the following keyword:

```
<Definition.of.Atomic.Species
  O    O7.0-s2p2d1       O_PBE19
  H    H7.0-s2p1         H_PBE19
Definition.of.Atomic.Species>
```

- O7.0 means O7.0.pao.
- -s2p2d1 means 2, 2, and 1 radial functions are allocated to s-, p-, and d-orbitals.
- In this case, for oxygen atom, 2×1+2×3+1×5=13 basis functions are allocated.
- O_PBE19 means O_PBE19.vps.

The path for O7.0.pao and O_PBE13.vps is specified by

DATA.PATH ../DFT_DATA19

Default value is '../DFT_DATA19'.

How to choose basis functions: H_2O case

By clicking H7.0.pao and O7.0.pao in the database(2019), you may find the following

https://www.openmx-square.org/vps_pao2019/H/index.html

H7.0.pao

```
**************************************************
  Eigen values(Hartree) of pseudo atomic orbitals
**************************************************

Eigenvalues
Lmax= 3 Mul=15
  | mu    0  0     -0.23595211038442   <- 3
  | mu    0  1      0.14109389991827   <- 6
  | mu    0  2      0.61751730037441
  | mu    0  3      1.31890671598573
  | mu    0  4      2.24052765608302
  | mu    0  5      3.37954791544661
  | mu    0  6      4.73488369825610
  | mu    0  7      6.30608874470710
  | mu    0  8      8.09282718517299
  | mu    0  9     10.09464035732420
  | mu    0 10     12.31085267019158
  | mu    0 11     14.74057314485273
  | mu    0 12     17.38277845742691
  | mu    0 13     20.23645090753857
  | mu    0 14     23.30073926597344   <- 5
  | mu    1  0      0.10914684890465
  | mu    1  1      0.47776040452236
  | mu    1  2      1.06988680483686
  | mu    1  3      1.88261331124981
  | mu    1  4      2.91175885838084
  | mu    1  5      4.15601184789448
  | mu    1  6      5.61454131060210
  | mu    1  7      7.28681796296307
  | mu    1  8      9.17254361476158
  | mu    1  9     11.27156766390586
  | mu    1 10     13.58381334569800
  | mu    1 11     16.10921637499960
  | mu    1 12     18.84767560575107
  | mu    1 13     21.79902110024685
  | mu    1 14     24.96300480798286
  | mu    2  0      0.27851528500170
  | mu    2  1      0.76970400958463
  | mu    2  2      1.47576814497054
  | mu    2  3      2.39881523735167
  | mu    2  4      3.53713346447754
```

https://www.openmx-square.org/vps_pao2019/O/index.html

O7.0.pao

```
**************************************************
  Eigen values(Hartree) of pseudo atomic orbitals
**************************************************

Eigenvalues
Lmax= 3 Mul=15
  | mu    0  0     -0.87913976280231   <- 1
  | mu    0  1      0.06809061901229   <- 4
  | mu    0  2      0.52709275865941
  | mu    0  3      1.24995140722317
  | mu    0  4      2.21829552402723
  | mu    0  5      3.41945468859267
  | mu    0  6      4.84509607059749
  | mu    0  7      6.48825090142865
  | mu    0  8      8.34207134316001
  | mu    0  9     10.39973244132192
  | mu    0 10     12.65513764955926
  | mu    0 11     15.10428688136984
  | mu    0 12     17.74677530362947
  | mu    0 13     20.58633940582683
  | mu    0 14     23.62957122674031   <- 2
  | mu    1  0     -0.33075182895384
  | mu    1  1      0.16376499567753   <- 7
  | mu    1  2      0.64129274864838
  | mu    1  3      1.35995471521821
  | mu    1  4      2.31377697411480
  | mu    1  5      3.50052618379026
  | mu    1  6      4.91841590421346
  | mu    1  7      6.56499793348396
  | mu    1  8      8.43634105410091
  | mu    1  9     10.52733540479795
  | mu    1 10     12.83276989780839
  | mu    1 11     15.34880171573570
  | mu    1 12     18.07414632417004
  | mu    1 13     21.01013110934429
  | mu    1 14     24.15933607166566   <- 8
  | mu    2  0      0.26162948257116
  | mu    2  1      0.70705247436937
  | mu    2  2      1.34714706672243
  | mu    2  3      2.19799459356269
  | mu    2  4      3.26511989658328
```

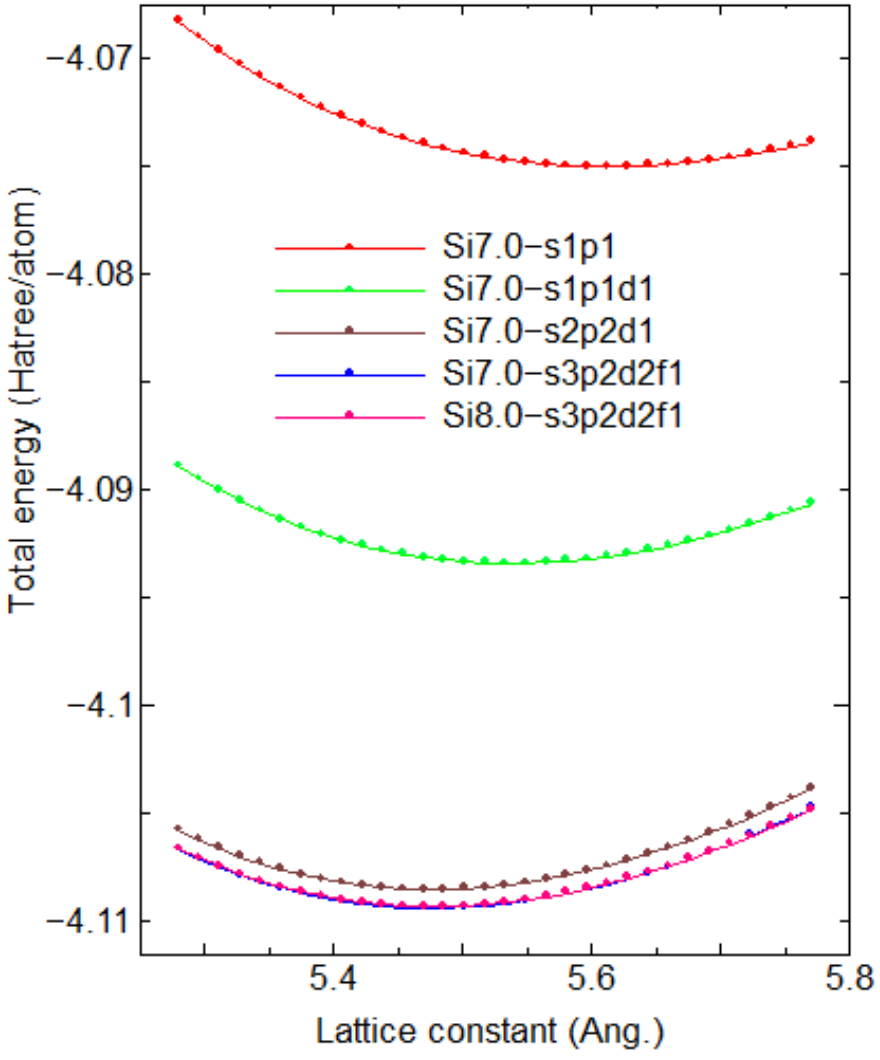
Choosing states with lower eigenvalues leads to H7.0-s2p1 and O7.0-s2p2d1.

How to choose basis functions: Si case(1)

Si7.0.pao

```
Eigenvalues
Lmax= 3 Mul=15
 | mu    0  0     -0.59320968623145 1
 | mu    0  1      0.01654247265872 4
 | mu    0  2      0.57915688461708 8
 | mu    0  3      1.39915528592998
 | mu    0  4      2.43106703909053
 | mu    0  5      3.63498884739274
 | mu    0  6      4.99885012002783
 | mu    0  7      6.55485759476897
 | mu    0  8      8.33541526908920
 | mu    0  9     10.34055461939939
 | mu    0 10     12.55818051232040
 | mu    0 11     14.98369777887096
 | mu    0 12     17.62014479571136
 | mu    0 13     20.47011746372617
 | mu    0 14     23.53245309073866
 | mu    1  0     -0.31460013133101 2
 | mu    1  1      0.12937640798489 5
 | mu    1  2      0.71637380083701
 | mu    1  3      1.54488697995006
 | mu    1  4      2.59211686526084
 | mu    1  5      3.84687324239366
 | mu    1  6      5.31246180826158
 | mu    1  7      6.99509799702661
 | mu    1  8      8.89454075291723
 | mu    1  9     11.00602102880752
 | mu    1 10     13.32692005826443
 | mu    1 11     15.85857455673626
 | mu    1 12     18.60272353142246
 | mu    1 13     21.55894843635971
 | mu    1 14     24.72602323954447
 | mu    2  0      0.01886411574821 3
 | mu    2  1      0.35893514996065 7
 | mu    2  2      0.94629918754692
 | mu    2  3      1.76201765644983
 | mu    2  4      2.81418723624388
 | mu    2  5      4.10656012645961
 | mu    2  6      5.63661971875011
 | mu    2  7      7.39522693820483
 | mu    2  8      9.37222098331867
 | mu    2  9     11.56227937764802
 | mu    2 10     13.96620568880690
 | mu    2 11     16.58651812641581
 | mu    2 12     19.42296574188659
 | mu    2 13     22.47308081363278
 | mu    2 14     25.73538272482773
 | mu    3  0      0.28356769151846 6
 | mu    3  1      0.79082836114569
 | mu    3  2      1.52992065308726
 | mu    3  3      2.52537261496132
```

Orbitals with lower eigenvalues in Si7.0.pao are taken into account step by step as the quality of basis set is improved.

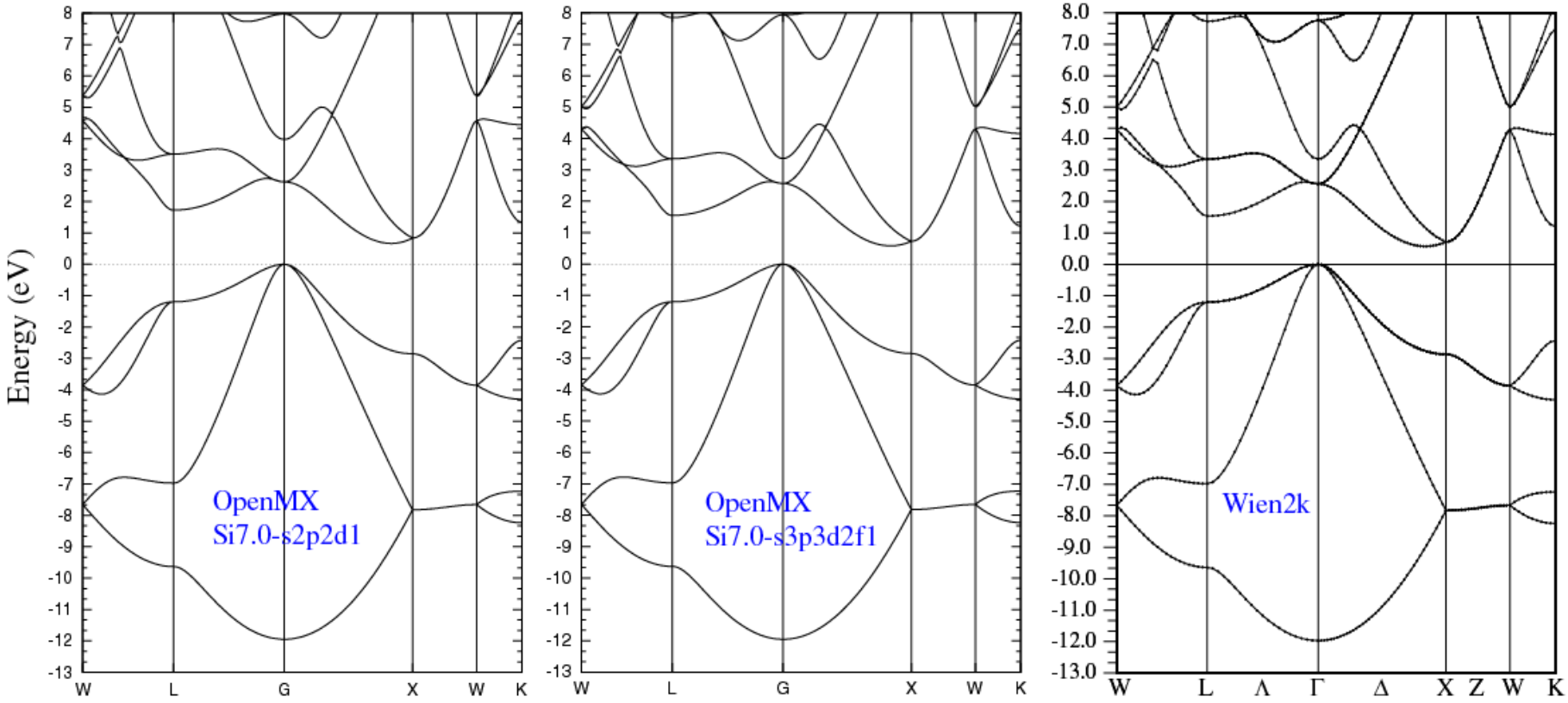


Si7.0-s2p2d1 is enough to discuss structural properties.

By comparing Si7.0-s3p2d2f1 with Si8.0-s3p2d2f1, it turns out that convergence is achieved at the cutoff of 7.0(a.u.).

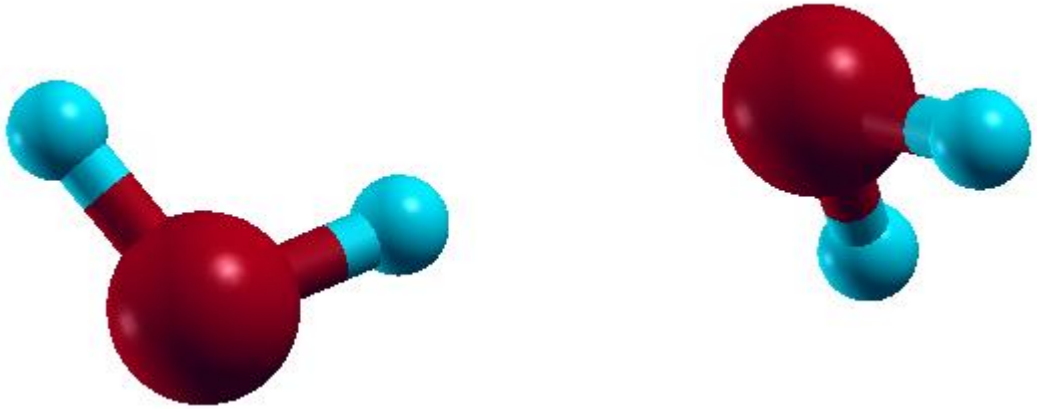
How to choose basis functions: Si case(2)

With respect to band structure, one can confirm that Si7.0-s2p2d1 provides a nearly convent result.



While the convergent result is achieved by use of Si7.0-s3p2d2f1(Si7.0-s3p3d2f1), Si7.0-s2p2d1 is a balanced basis functions compromising accuracy and efficiency to perform a vast range of materials exploration.

Basis set superposition Error (BSSE)



	Equilibrium O-O distance (Ang.)	Dipole moment (Debye)	Binding energy (kcal/mol)	Binding energy (couterpoise corrected) (kcal/mol)
O7.0-s2p2d1, H7.0-s2p1	2.899	2.54	5.57	5.21
O7.0-s3p3d2, H7.0-s3p2	2.897	2.45	5.48	5.48
Other calc.	2.893[a]		5.15 [a]	
Expt.	2.98[b]	2.60[b]	5.44 [b]	

https://www.openmx-square.org/vps_pao2019/O/index.html

A series of benchmark calculations implies that BSSE is ～0.5 kcal/mol for molecular systems.

Recommendation of choices for PAO

VPS	Valence electrons	Quick	Standard	Precise
E	0.0	Kr10.0-s1p1	Kr10.0-s2p1d1	Kr10.0-s2p2d1f1
H_PBE19	1.0	H5.0-s2	H6.0-s2p1	H7.0-s2p2d1
He_PBE19	2.0	He8.0-s1p1	He8.0-s2p1	He10.0-s2p2d1
Li_PBE19	3.0	Li8.0-s3p1	Li8.0-s3p2	Li8.0-s3p2d1
Be_PBE19	2.0	Be7.0-s2p1	Be7.0-s2p2	Be7.0-s3p2d1
B_PBE19	3.0	B7.0-s2p2	B7.0-s2p2d1	B7.0-s3p2d2
C_PBE19	4.0	C6.0-s2p2	C6.0-s2p2d1	C6.0-s3p2d2
N_PBE19	5.0	N6.0-s2p2	N6.0-s2p2d1	N6.0-s3p2d2
O_PBE19	6.0	O6.0-s2p2	O6.0-s2p2d1	O6.0-s3p2d2
F_PBE19	7.0	F6.0-s2p2	F6.0-s2p2d1	F6.0-s3p3d2f1
Ne_PBE19	8.0	Ne9.0-s2p2	Ne9.0-s2p2d1	Ne9.0-s3p2d2
Na_PBE19	9.0	Na9.0-s3p2	Na9.0-s3p2d1	Na9.0-s3p2d2
Mg_PBE19	8.0	Mg9.0-s2p2	Mg9.0-s3p2d1	Mg9.0-s3p2d2
Al_PBE19	3.0	Al7.0-s2p1d1	Al7.0-s2p2d1	Al7.0-s3p2d2
Si_PBE19	4.0	Si7.0-s2p1d1	Si7.0-s2p2d1	Si7.0-s3p3d2
P_PBE19	5.0	P7.0-s2p2d1	P7.0-s2p2d1f1	P7.0-s3p2d2f1
S_PBE19	6.0	S7.0-s2p2d1	S7.0-s2p2d1f1	S7.0-s3p2d2f1
Cl_PBE19	7.0	Cl7.0-s2p2d1	Cl7.0-s2p2d1f1	Cl7.0-s3p2d2f1

See https://www.openmx-square.org/openmx_man3.9/node27.html

Structure of a molecule

of atoms

Unit of atomic coordinate

```
Atoms.Number       5
Atoms.SpeciesAndCoordinates.Unit   Ang # Ang|AU|FRAC
<Atoms.SpeciesAndCoordinates
 1  C     0.000000   0.000000   0.000000    2.0  2.0
 2  H    -0.889981  -0.629312   0.000000    0.5  0.5
 3  H     0.000000   0.629312  -0.889981    0.5  0.5
 4  H     0.000000   0.629312   0.889981    0.5  0.5
 5  H     0.889981  -0.629312   0.000000    0.5  0.5
Atoms.SpeciesAndCoordinates>
```

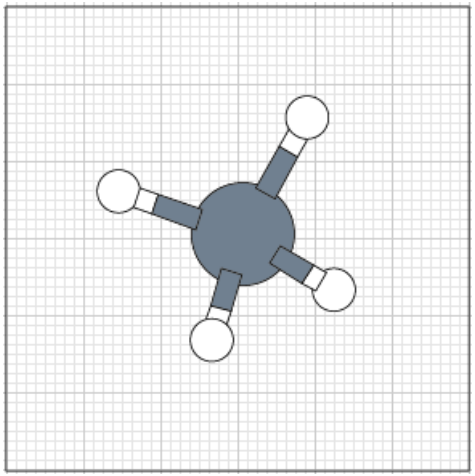
Initial occupation for up and down states of each atom.
See also the manual;
https://www.openmx-square.org/openmx_man3.9/node27.html

Atomic coordinates

```
Atoms.UnitVectors.Unit         Ang # Ang|AU
<Atoms.UnitVectors
  10.0   0.0   0.0
   0.0  10.0   0.0
   0.0   0.0  10.0
Atoms.UnitVectors>
```

Unit of unit vectors

Unit vectors
a1
a2
a3



Structure of a bulk

```
Atoms.Number                    2
Atoms.SpeciesAndCoordinates.Unit   FRAC
<Atoms.SpeciesAndCoordinates
  1     Si   0.7500000   0.7500000   0.7500000   2.0  2.0
  2     Si   0.5000000   0.5000000   0.5000000   2.0  2.0
Atoms.SpeciesAndCoordinates>

Atoms.UnitVectors.Unit          Ang
<Atoms.UnitVectors
  3.8669746   0.0000000   0.0000000
  1.9334873   3.3488983   0.0000000
  1.9334873   1.1162994   3.1573716
Atoms.UnitVectors>
```

# of atoms

Unit of atomic coordinate

Initial occupation for up and down states of each atom. See also the manual;

https://www.openmx-square.org/openmx_man3.9/node27.html

Atomic coordinates

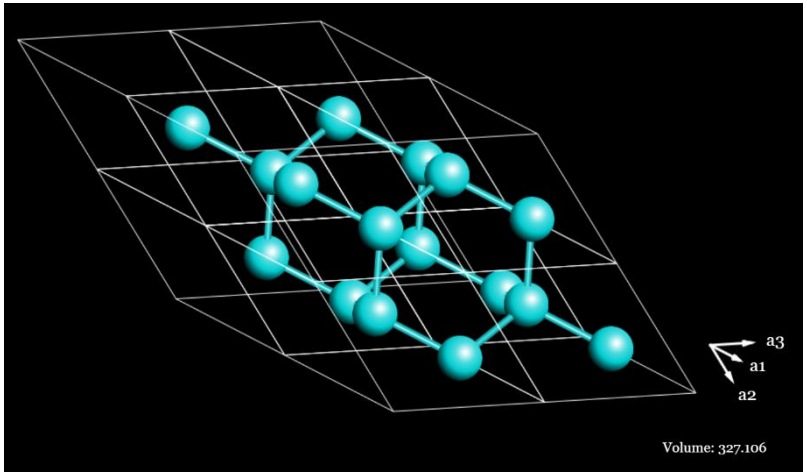
Unit of unit vectors

Unit vectors

a1

a2

a3



Choice of cutoff energy

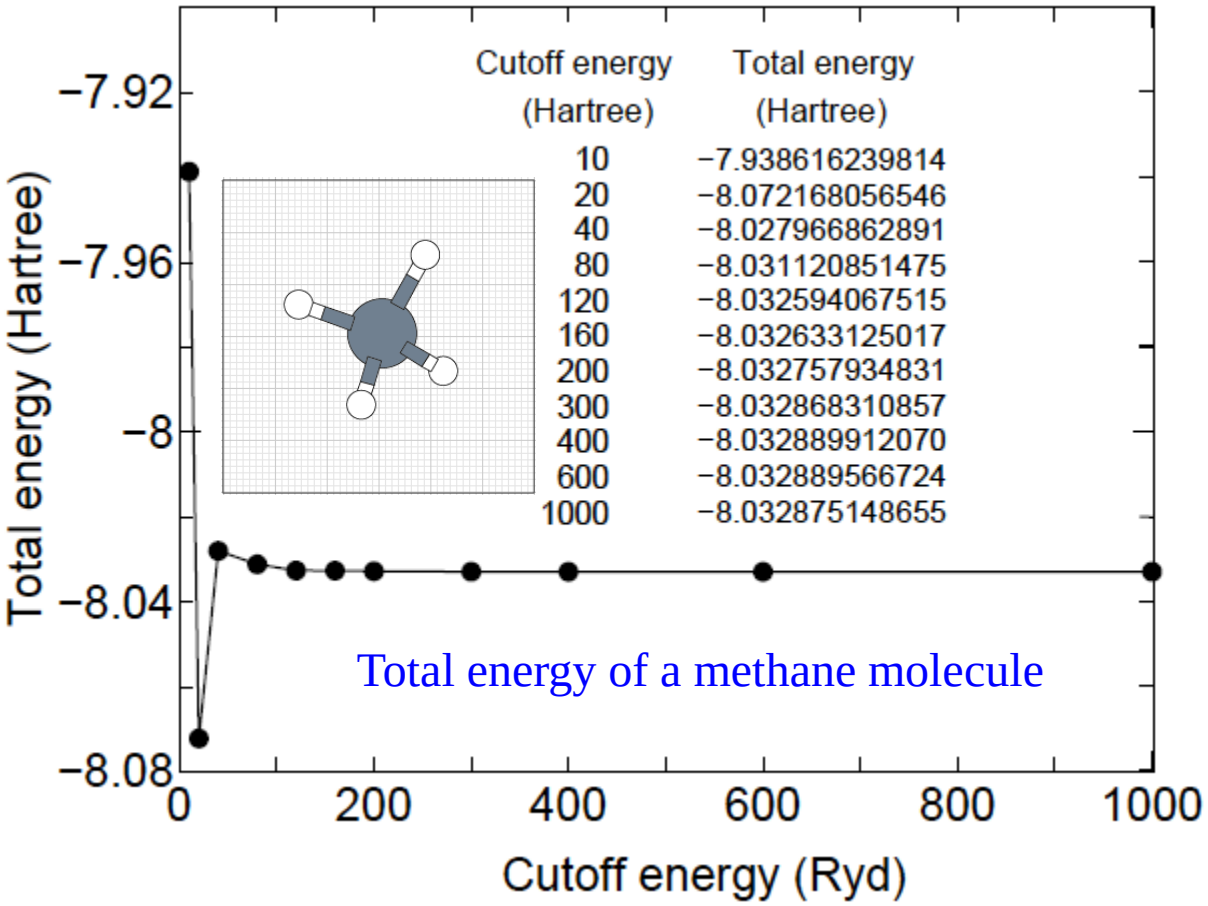
scf.energycutoff 200 # default=150 Ryd

The FFT grid is used to discretize real space and calculate $E_{\delta ee}$, E_{xc}, and can be specified by scf.energycutoff.

In most cases, 200 Ryd is enough to get convergence.

However, large cutoff energy (300-400 Ryd) has be used cases such as use of pseudopotentials with deep semi-core states.

Memory requiment
$O(E^{3/2})$



Choice of cutoff energy

Geometry optimization of H_2O

Dependency of optimized structure of H_2O on scf.energycutoff. It turns out that 180Ryd. is enough to reach the convergence.

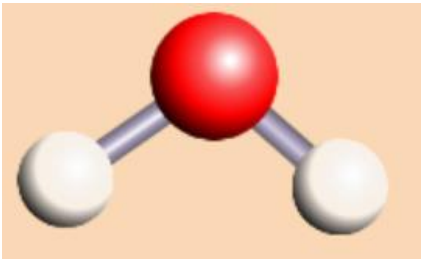


Table 1: Convergence of structural parameters, dipole moment of a water molecule with respect to the cutoff energy. The input file is 'H2O.dat' in the directory 'work'.

Ecut(Ryd)	r(H-O) (Å)	∠ (H-O-H) (deg)	Dipole moment (Debye)
60	0.970	103.4	1.838
90	0.971	103.7	1.829
120	0.971	103.7	1.832
150	0.971	103.6	1.829
180	0.971	103.6	1.833
Exp.	0.957	104.5	1.85

Volume vs. Energy curves

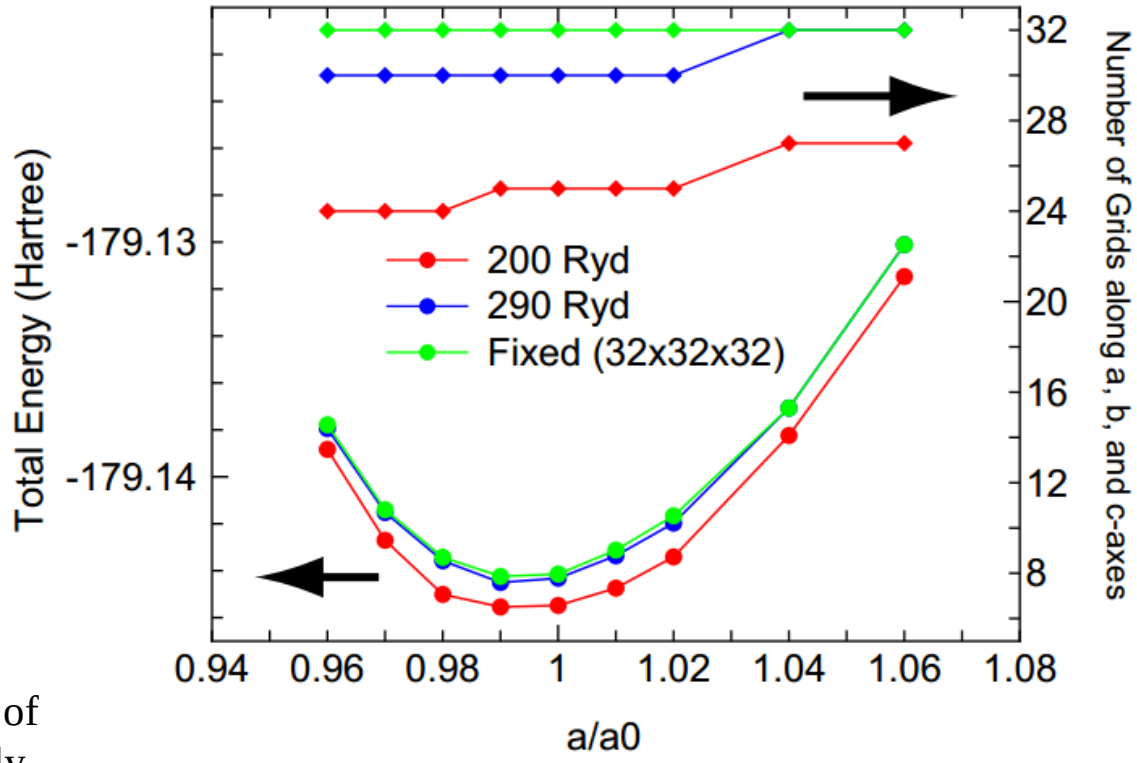
The following keywords are available to calculate energy curves.

```
MD.Type          EvsLC    #
MD.EvsLC.Step    0.4      # default=0.4%
MD.maxIter       32       # default=1
```

When the energy curve for bulk system is calculated as a function of the lattice parameter, a sudden change of the number of real space grids is a serious problem which produces an erratic discontinuity on the energy curve. To avoid this, the number of grids should be fixed by explicitly specifying the following keyword:

```
scf.Ngrid        32 32 32
```

The numbers correspond to the number of grid along a-, b-, and c-axes, respectively. scf.Ngrid is used if both the keywords scf.energycutoff and scf.Ngrid are specified.



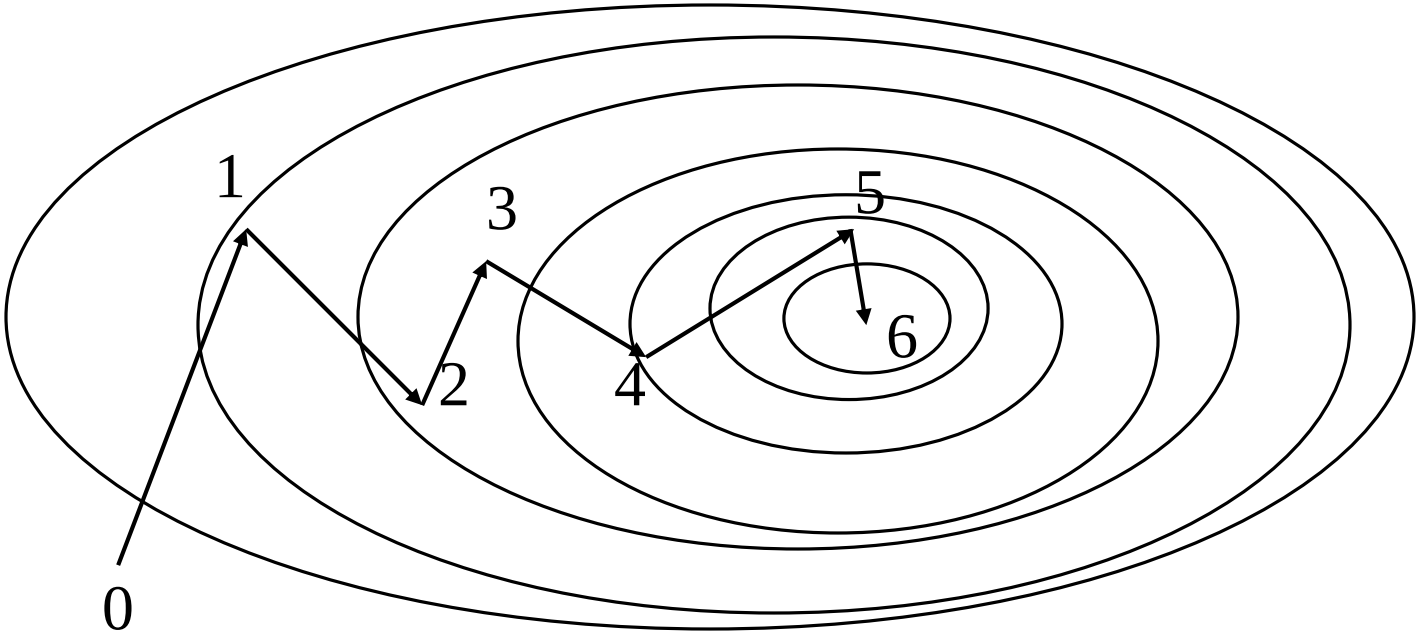

```
MD.EvsLC.flag    1  1  0      # along a, b, c-axes
1: varied,  0: fixed
```

Geometry optimization

To quantitatively investigate structural, physical, and chemical properties of molecules and solids, it would be important to obtain optimized structures.

Steepest decent (SD) method

The structure is changed along the steepest decent step by step.



It may not be so efficient.

Geometry optimization by the SD method

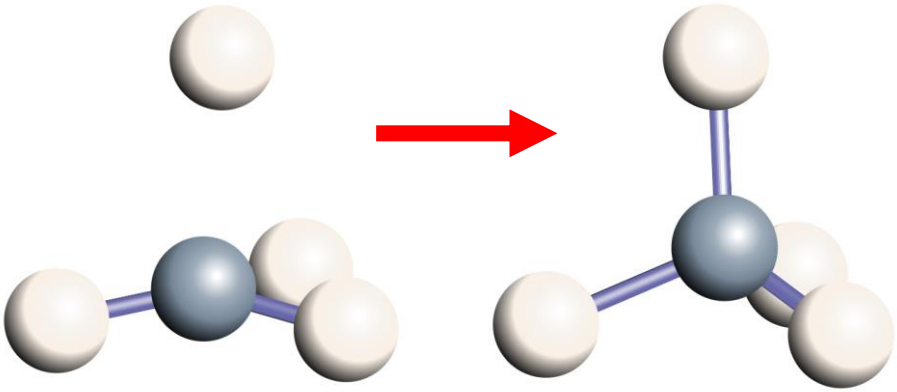
Let us change the *x*-coordinate of carbon atom in a methane molecule to 0.3 Å as

```
<Atoms.SpeciesAndCoordinates
   1   C     0.300000    0.000000    0.000000     2.0  2.0
   2   H    -0.889981   -0.629312    0.000000     0.5  0.5
   3   H     0.000000    0.629312   -0.889981     0.5  0.5
   4   H     0.000000    0.629312    0.889981     0.5  0.5
   5   H     0.889981   -0.629312    0.000000     0.5  0.5
 Atoms.SpeciesAndCoordinates>
```

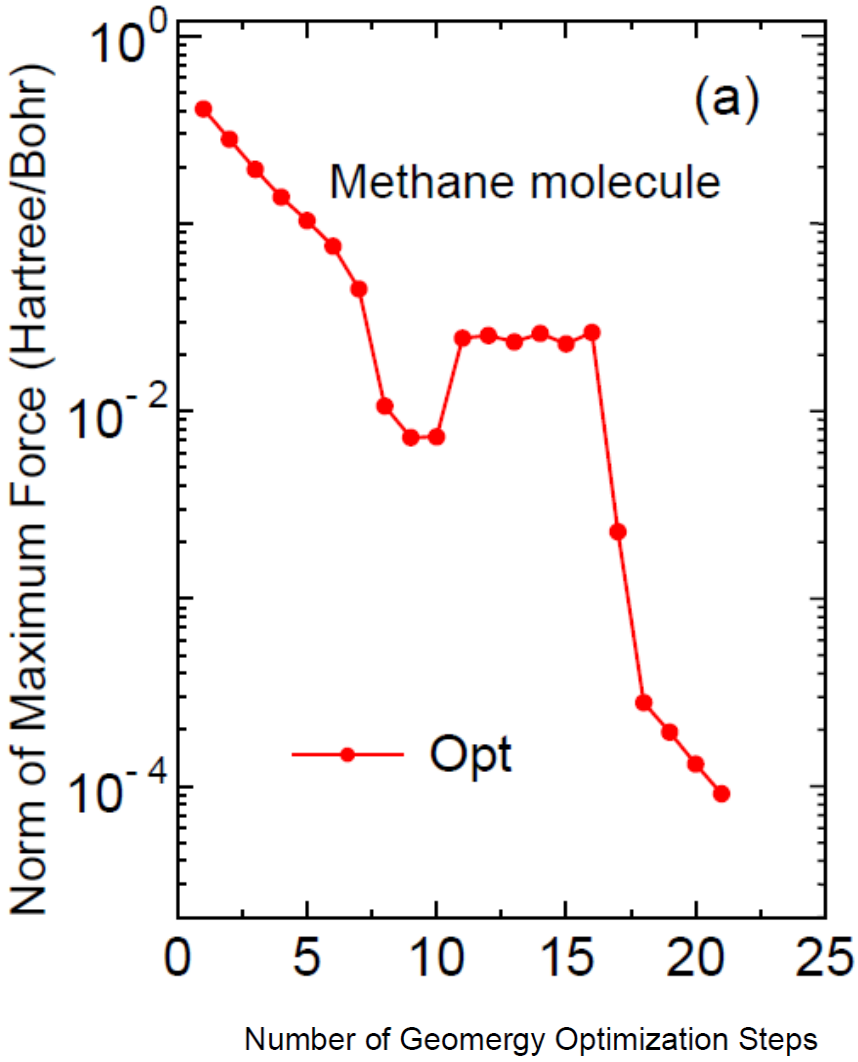
Using “Methane2.dat” in the directory “work”, you can trace the calculation.

Initial structure

Final structure



Optimization process



Relevant output files for the geometry optimization

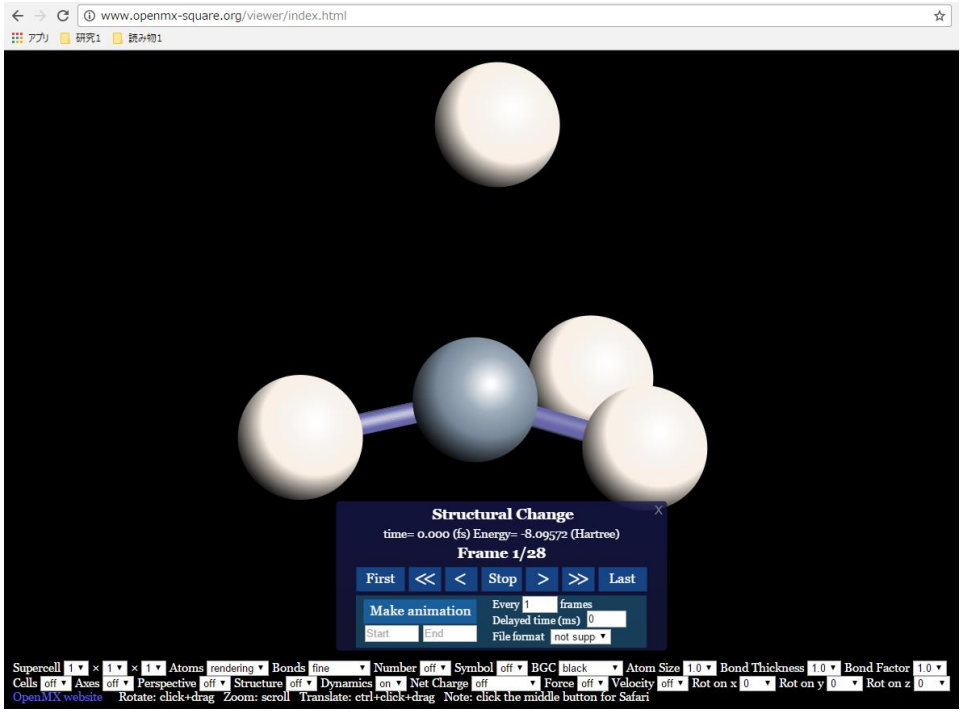
In met2.out, the history of optimization can be confirmed.

By dragging and dropping met2.md to OpenMX Viewer, the optimization process can be easily visualized.

```
************************************************************
************************************************************
             History of geometry optimization
************************************************************
************************************************************

  MD_iter    SD_scaling      |Maximum force|    Maximum step        Utot
                             (Hartree/Bohr)        (Ang)          (Hartree)

     1       1.25981733        0.40873710       0.10583545      -8.09571722
     2       1.25981733        0.12000148       0.08000099      -8.16523842
     3       1.25981733        0.06115237       0.04076824      -8.18400222
     4       1.25981733        0.01811189       0.01207460      -8.18831175
     5       3.14954331        0.01076505       0.01794175      -8.18898468
     6       3.14954331        0.00576811       0.00961351      -8.18980390
     7       3.14954331        0.00355034       0.00591724      -8.19010991
     8       3.14954331        0.00255760       0.00426266      -8.19022887
     9       7.87385828        0.00218904       0.00912099      -8.19027446
    10       7.87385828        0.00863395       0.03597478      -8.19023586
    11       1.57477166        0.04692594       0.03910495      -8.18816953
    12       1.57477166        0.01357319       0.01131099      -8.19012703
    13       3.93692914        0.00406864       0.00847634      -8.19029047
    14       3.93692914        0.00915707       0.01907723      -8.19022710
    15       0.78738583        0.02067017       0.00861257      -8.18990108
    16       0.78738583        0.00721609       0.00300671      -8.19025699
    17       1.96846457        0.00255359       0.00265999      -8.19030094
    18       1.96846457        0.00163789       0.00170613      -8.19030460
    19       1.96846457        0.00105501       0.00109897      -8.19030613
    20       1.96846457        0.00068667       0.00071528      -8.19030671
    21       4.92116143        0.00044434       0.00115713      -8.19030697
    22       4.92116143        0.00138402       0.00360421      -8.19030537
    23       0.98423229        0.00427346       0.00222576      -8.19028994
    24       0.98423229        0.00078570       0.00040922      -8.19030656
    25       2.46058071        0.00013628       0.00017745      -8.19030709
    26       2.46058071        0.00014411       0.00018765      -8.19030714
    27       0.49211614        0.00015118       0.00003937      -8.19030709
    28       0.49211614        0.00008899       0.00002317      -8.19030713
```



Geometry optimization by Newton-type methods

By Taylor-expanding, we have

$$E = E_0 + \sum_i^{3N} \left(\frac{\partial E}{\partial x_i}\right)_0 (x_i - x_i^{(0)}) + \frac{1}{2}\sum_{i,j}^{3N} \left(\frac{\partial^2 E}{\partial x_i \partial x_j}\right)_0 (x_i - x_i^{(0)})(x_j - x_j^{(0)}) + \cdots,$$

The derivative of the total energy w.r.t. coordinates leads to

$$\frac{\partial E}{\partial x_k} = \left(\frac{\partial E}{\partial x_k}\right)_0 + \sum_i^{3N} \left(\frac{\partial^2 E}{\partial x_k \partial x_i}\right)_0 (x_i - x_i^{(0)}).$$

$$\frac{\partial E}{\partial x_k} = 0 \longrightarrow \begin{pmatrix} (\frac{\partial^2 E}{\partial x_1 \partial x_1})_0 & (\frac{\partial^2 E}{\partial x_1 \partial x_2})_0 & \cdots \\ (\frac{\partial^2 E}{\partial x_2 \partial x_1})_0 & (\frac{\partial^2 E}{\partial x_2 \partial x_2})_0 & \cdots \\ \cdots & \cdots & \cdots \end{pmatrix} \begin{pmatrix} (x_1 - x_1^{(0)}) \\ (x_2 - x_2^{(0)}) \\ \cdots \end{pmatrix} = - \begin{pmatrix} \left(\frac{\partial E}{\partial x_1}\right)_0 \\ \left(\frac{\partial E}{\partial x_2}\right)_0 \\ \cdots \end{pmatrix}.$$

$$H\Delta\mathbf{x} = -\mathbf{g},$$

By solving the Eq. for Δx, we have

$$\mathbf{x}^{(n+1)} = \mathbf{x}^{(n)} - (H^{(n)})^{-1}\mathbf{g}^{(n)}.$$

If the Hessian matrix H can be computed, the method is efficient.
However, it may be difficult to evaluate H in general.

Geometry optimization by Newton-type methods

The geometry optimization in OpenMX is based on quasi Newton methods. In Ver. 3.9, the following four methods are available.

$$\mathbf{r}_{\rm new} = \mathbf{r}_{\rm DIIS} + \Delta\mathbf{r} \qquad\qquad \Delta\mathbf{r} = -H^{-1}\mathbf{g}_{\rm DIIS}$$

Methods of calculating approximate Hessian matrix H

DIIS	BFGS	RF(rational function)	EF(eigenvector following)
H=I	BFGS	BFGS+RF	BFGS plus monitoring of eigenvalues of H

Broyden-Fletcher-Goldfarb-Shanno (BFGS) method

$$H_k = H_{k-1} + \frac{|\Delta\mathbf{g}_k\rangle\langle\Delta\mathbf{g}_k|}{\langle\Delta\mathbf{g}_k|\Delta\mathbf{r}_k\rangle} - \frac{|H_k\Delta\mathbf{r}_k\rangle\langle\Delta\mathbf{r}_k H_k|}{\langle\Delta\mathbf{r}_k|H_k|\Delta\mathbf{r}_k\rangle}$$

If the inner product in the red box is positive, the positive definiteness of H is guaranteed.

Approximate initial Hessian by Schlegel

Schlegel proposed a way of constructing an approximate Hessian. A force constant for every pair of elements is fitted to the following formula, where dataset were constructed by B3LYP calculations.

$$F = \frac{A}{(r - B)^3}$$

H.B. Schlegel, Theoret. Chim. Acta (Berl.) 66, 333 (1984); J.M. Wittbrodt and H.B. Schlegel, J. Mol. Struc. (Theochem) 398-399, 55 (1997).

Parameter *B* for Badger's rule computed at the B3LYP level of theory

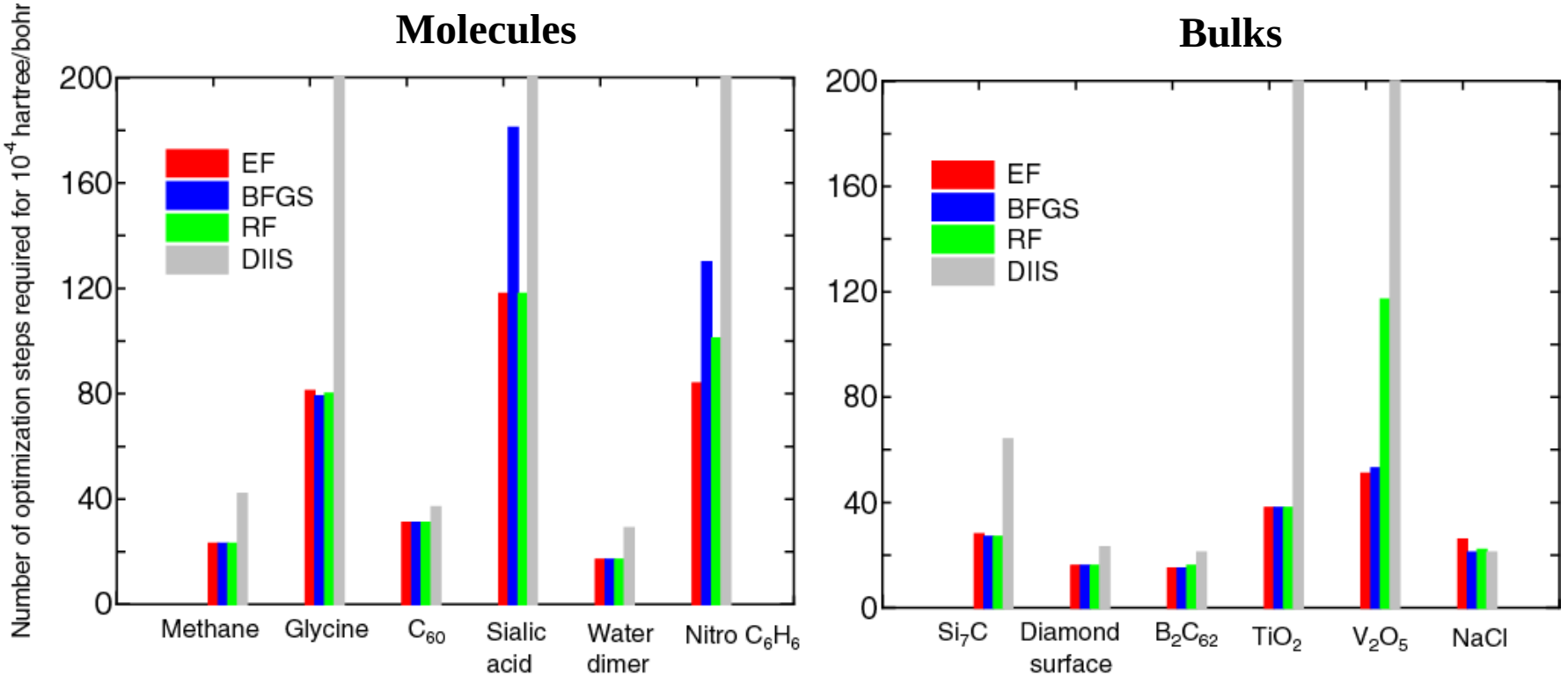
Period	1 H	2 Li–F	3 Na–Cl	4 K–Br	5 Rb–I	6 Cs–At
1	– 0.2573	0.3401	0.6937	0.7126	0.8335	0.9491
2		0.9652	1.2843	1.4725	1.6549	1.7190
3			1.6925	1.8238	2.1164	2.3185
4				2.0203	2.2137	2.5206
5					2.3718	2.5110

Suppose the total energy is given by the sum of pairwise potentials. Then, the derivatives lead to the following relation:

$$V_2 = \frac{1}{2}\sum_i \left(\sum_{Rn} \sum_j f(|r_i + R_n - r_j|) \right) \qquad H = BF$$

where B is the B-matrix of Wilson, H is the approximate Hessian in Cartesian coordinate.

Comparison of four methods



- The benchmark calculations imply that the EF and RF work well.
- Large molecules with structural large freedom are hard to get convergence.

The input file and output files for the benchmark calculations are available in openmx3.9/work/geoopt_example".

Keywords relevant to geometry optimization

```
MD.Type                 EF      # Opt|DIIS|BFGS|RF|EF
MD.Opt.DIIS.History      3      # default=3
MD.Opt.StartDIIS         5      # default=5
MD.Opt.EveryDIIS       200      # default=200
MD.maxIter             100      # default=1
MD.Opt.criterion       1.0e-4   # default=0.0003 (Hartree/Bohr)
```

The behavior of the quasi Newton methods can be controlled by the following two keywords:

```
MD.Opt.DIIS.History     3 # default=3
MD.Opt.StartDIIS        5 # default=5
```

The keyword 'MD.Opt.DIIS.History' gives the number of previous steps to estimate the optimized structure used in the geometry optimization by 'DIIS', 'EF', and 'RF'. The default value is 3.

The geometry optimization step at which 'DIIS', 'EF', or 'RF' starts is specified by the keyword 'MD.Opt.StartDIIS'. The geometry optimization steps before starting the DIIS-type method is performed by the steepest decent method. The default value is 5.

https://www.openmx-square.org/openmx_man3.9/node47.html

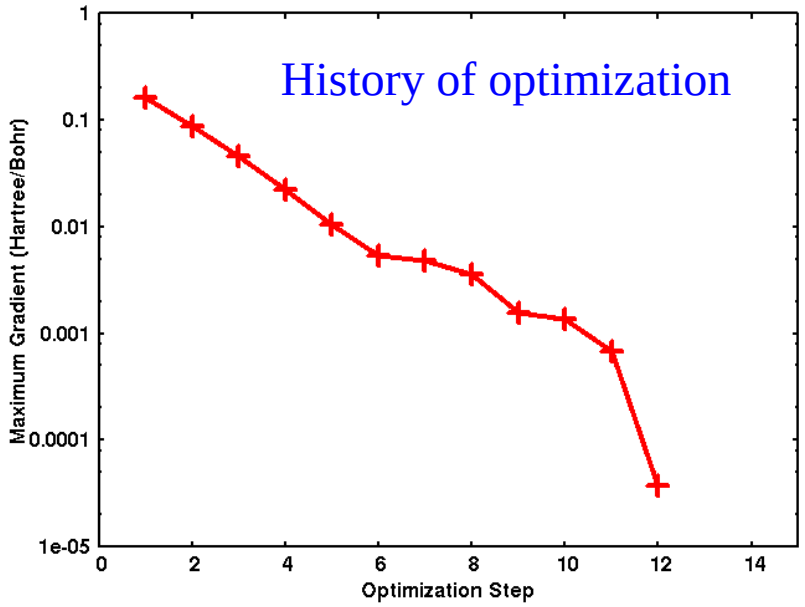
Variable cell optimization

Let us start optimization of diamond lattice with a displacement as

```
Atoms.Number          2
Atoms.SpeciesAndCoordinates.Unit   frac # Ang|AU
<Atoms.SpeciesAndCoordinates
   1     C     0.10000000000000    0.00000000000000   -0.05000000000000      2.0      2.0
   2     C     0.25000000000000    0.25000000000000    0.25000000000000      2.0      2.0
Atoms.SpeciesAndCoordinates>
Atoms.UnitVectors.Unit              Ang # Ang|AU
<Atoms.UnitVectors
  1.6400  1.6400  0.0000
  1.6400  0.0000  1.6400
  0.0000  1.6400  1.6400
Atoms.UnitVectors>
```

Relevant keywords:

```
MD.Type                          RFC5
MD.Opt.DIIS.History               3
MD.Opt.StartDIIS                  7
MD.Opt.EveryDIIS                  100000
MD.maxIter                        100
MD.TimeStep                      1.0
MD.Opt.criterion                0.0003
```

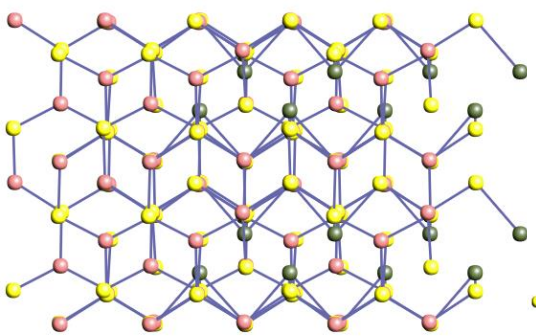


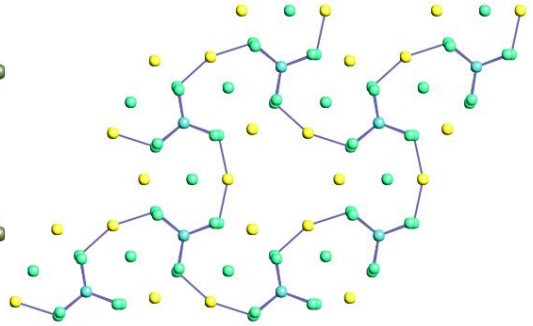
The calculation can be traced by “Cdia-RF5.dat” in work/cellopt_example.

Please see also the page 74 in the manual.

Benchmark calculations of RFC5

For 785 crystals (mostly sulfides) , the full optimization by RFC5 were performed by Mr. Miyata, Ph.D student in JAIST, as computational screening in searching good thermoelectric materials





The optimization criterion: 10^{-4} Hartree/bohr

The histogram shows the number of systems among 785 systems as a function of the number of iterations to achieve the convergence

Restarting of calculations

- After finishing your first calculation or achieving the self consistency, you may want to continue the calculation or to calculate density of states, band dispersion, molecular orbitals, and etc. using the self consistent charge in order to save the computational time. To do this, a keyword 'scf.restart' is available.

```
scf.restart        on        # on|off,default=off
```

- If the first trial for geometry optimization does not reach a convergent result or molecular a dynamics simulation is terminated due to a wall time, one can restart the geometry optimization using an input file 'System.Name.dat#' which is generated at every step for the restart calculation with the final structure.

See also the manual at

https://www.openmx-square.org/openmx_man3.9/node44.html

Output of large-sized files in binary mode

Large-scale calculations produce large-sized files in text mode such as cube files. The IO access to output such files can be very time consuming in machines of which IO access is not fast. In such a case, it is better to output those large-sized files in binary mode. The procedure is supported by the following keyword:

```
OutData.bin.flag    on    # default=off, on|off
```

Then, all large-sized files will be output in binary mode. The default is 'off'.
The output binary files are converted using a small code 'bin2txt.c' stored in the directory 'source' which can be compiled as

```
gcc bin2txt.c -lm -o bin2txt
```

As a post processing, you will be able to convert as

```
./bin2txt *.bin
```

The functionality will be useful for machines of which IO access is not fast.

See also the manual: https://www.openmx-square.org/openmx_man3.9/node218.html