

Getting Started with OpenMX Ver. 3.3

1. Get the package and its installation
2. Running of test jobs
3. Input files
4. Output files
5. Basis sets and pseudopotentials
6. Functionals
7. Cutoff energy for regular mesh
8. Getting SCF
9. Geometry optimization
10. Initial spin moments
11. Visualization of cube files

The PDF file

is available at

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

Get the package

(1) Go to

<http://www.openmx-square.org/download.html>

(2) Download the Ver.3.3 and its patch



Download of OpenMX

Available packages in terms of [GNU-GPL](#)

- [openmx3.3 \(release date: 30/July/2007\)](#)
 - [openmx3.2 \(release date: 01/Apr./2007\)](#)
 - [openmx3.1 \(release date: 04/Nov./2006\)](#)
 - [openmx3.0 \(release date: 20/Mar./2006\)](#)
- + [patch \(25/Sep./2007\)](#)
+ [patch \(02/July/2007\)](#)

Installation No.0

Before the installation of OpenMX, the installation of **LAPACK, BLAS, FFTW** is indispensable.

Of course, **variants** of LAPACK/BLAS are also **available** instead, such as

ATLAS,
ACML for AMD machines
MKL for Intel machines

Installation No.1

(1) Decompress the tar file

```
Xkterm  
[ozaki@vtppcc01 ozaki]$ tar zxvf openmx3.3.tar.gz █
```

(2) Decompress the patch

```
Xkterm  
[ozaki@vtppcc01 ozaki]$  
[ozaki@vtppcc01 ozaki]$ cp patch3.3.1.tar.gz openmx3.3/source/  
[ozaki@vtppcc01 ozaki]$ cd openmx3.3/source/  
[ozaki@vtppcc01 source]$ tar zxvf patch3.3.1.tar.gz  
openmx_common.h  
Set_Vpot.c  
Total_Energy.c  
[ozaki@vtppcc01 source]$ █
```

(3) You will find three directories below openmx3.3

```
Xkterm  
[ozaki@vtppcc01 ozaki]$ cd openmx3.3  
[ozaki@vtppcc01 openmx3.3]$ ls  
DFT_DATA source work  
[ozaki@vtppcc01 openmx3.3]$ █
```

Installation No.2

(1) Modify makefile in openmx3.3/source

```
Xkterm
```

```
[ozaki@vtppcc01 ozaki]$ cd openmx3.3/source/  
[ozaki@vtppcc01 source]$ emacs -nw makefile █
```

```
# HA8000 (Xeon in IMS)  
#  
# CC      = mpicc -nocheckpoint -compiler intel17 -Dblaswrap -O3 -I/home/o000/nak1/t  
# LIB     = -L/home/o000/nak1/HA/gcc/lib -lfftw3 -llapack -lblas -lg2c -li77 -stati  
#
```

```
CC      = /usr/local/mpich-1.2.5-intel/bin/mpicc -O3 -DTRAN -I/usr/local/include -I  
LIB     = -L/home/ozaki/lib -latlas_p4 -lfftw3 -static
```

```
CFLAGS = -g
```

The two lines “CC” and “LIB” must be changed depending on your computational environment.

(2) Compile the sources

```
Xkterm
```

```
[ozaki@vtppcc01 source]$ make install  
/usr/local/mpich-1.2.5-intel/bin/mpicc -O3 -DTRAN -I/usr/local/inclur  
/usr/local/mpich-1.2.5-intel/bin/mpicc -O3 -DTRAN -I/usr/local/inclur
```

Installation No.3

Compiler options (not all, see the manual for more details)

-Dnomp1 for the serial version

-Dfftw2 for use of FFTW2

Installation No.4

You can find several examples for setting CC and LIB in makefile.

```
#####  
#  
# Please set a proper CC and LIB for the compilation.  
# Examples of CC and LIB on several platforms are shown below.  
#  
#####  
#  
# Cygwin, IBM ThinkPAD X40 (Pentium M 1.0GHz)  
#  
# CC = gcc -Dnomp1 -Dblaswrap -O3 -I/home/ozaki/include  
# STACK = -Wl,--heap,9000000,--stack,9000000  
# LIB =-L/home/ozaki/lib -llapack -lblas -lg2c -li77 -lfftw3 -static
```

Also, tips for installation can be found in the OpenMX Forum as

▲ Tips for installation of OpenMX

Date: 2007/09/25 17:12
Name: **T.Ozaki**

Dear All,

The following is tips for installation of OpenMX.

Most problems in installation of OpenMX come from compilation of LAPACK and BLAS and its linking. Thus, I shall show tips for installation of OpenMX on several platforms.

(1) Intel Pentium 4 and Xeon (32 bit)

Running of test jobs No.1

Move to the directory 'work', and perform the program, openmx, using an input file, *Methane.dat*, which can be found in the directory 'work' as follows:

```
Xkterm  
[ozaki@vtppcc01 work]$ pwd  
/home/ozaki/openmx3.3/work  
[ozaki@vtppcc01 work]$ ./openmx Methane.dat > met.std &
```

For the MPI version, perform as

```
Xkterm  
[ozaki@vtppcc01 work]$ pwd  
/home/ozaki/openmx3.3/work  
[ozaki@vtppcc01 work]$ mpirun -np 1 ./openmx Methane.dat > met.std &
```

Running of test jobs No.2

After the calculation, **ten files** and **one directory** will be 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.memory0	analysis for used memory
met.md	geometrical structures at every MD step
met.md2	geometrical structure of the final MD step
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_rst/	directory storing restart files

Running of test jobs No.3

met.out gives the standard information about the calculation such as the SCF convergence history, total energy, Mulliken charge, etc.

```
*****
*****
***** SCF history at MD= 1 *****
*****
SCF= 1 NormRD= 1.000000000000 Uele= -3.799184455669
SCF= 2 NormRD= 0.294505008903 Uele= -3.180922897246
SCF= 3 NormRD= 0.088735676388 Uele= -3.371991825409
SCF= 4 NormRD= 0.021096019843 Uele= -3.453330358005
SCF= 5 NormRD= 0.006019683678 Uele= -3.449516183258
SCF= 6 NormRD= 0.001743319193 Uele= -3.453580408828
SCF= 7 NormRD= 0.000014474996 Uele= -3.453268543024
SCF= 8 NormRD= 0.000006148039 Uele= -3.453265887480
SCF= 9 NormRD= 0.000004516849 Uele= -3.453266096345
SCF= 10 NormRD= 0.000003762514 Uele= -3.453266201895
SCF= 11 NormRD= 0.000003979367 Uele= -3.453266170692
SCF= 12 NormRD= 0.000000433153 Uele= -3.453266674615
SCF= 13 NormRD= 0.000000590053 Uele= -3.453266680973

*****
***** Total energy (Hartree) at MD = 1 *****
*****
Uele. -3.453266680973
Ukin. 5.824571431429
UH0. -14.517598400156
UH1. 0.012112580304
Una. -6.365977489798
Uh1. 0.681047544947
Uxc0. -1.609135573912
Uxc1. -1.609135573912
Ucore. 9.551521413583
Uhub. 0.000000000000
Ucs. 0.000000000000
Uzs. 0.000000000000
Uzo. 0.000000000000
Uef. 0.000000000000
Utot. -8.032594067514
```

Automatic running test

To check whether most functionalities of OpenMX have been successfully installed, it is possible to perform an automatic running test.

For the serial case

```
Xkterm  
[ozaki@vtppcc01 work]$ ./openmx -runtest
```

For the MPI case

```
Xkterm  
[ozaki@vtppcc01 |]$ mpirun -np 6 openmx -runtest
```

The result can be found in a file 'runtest.result' as

```
Xkterm  
[ozaki@vtppcc01 work]$ more runtest.result  
1 input_example/Benzene.dat Elapsed time(s)= 26.80 diff Utot= 0.000000000000 diff Force= 0.000000000000  
2 input_example/C60.dat Elapsed time(s)= 126.07 diff Utot= 0.000000000008 diff Force= 0.000000000001  
3 input_example/CO.dat Elapsed time(s)= 113.62 diff Utot= 0.000000002951 diff Force= 0.000000000000  
4 input_example/Cr2.dat Elapsed time(s)= 61.11 diff Utot= 0.000000000000 diff Force= 0.000000000000  
5 input_example/Crys-MnO.dat Elapsed time(s)= 187.31 diff Utot= 0.000000000217 diff Force= 0.000000000033  
6 input_example/GaAs.dat Elapsed time(s)= 191.91 diff Utot= 0.000000000074 diff Force= 0.000000000033  
7 input_example/Glycine.dat Elapsed time(s)= 40.89 diff Utot= 0.000000000000 diff Force= 0.000000000000  
8 input_example/Graphite4.dat Elapsed time(s)= 19.71 diff Utot= 0.000000000000 diff Force= 0.000000000000  
9 input_example/H2O-EF.dat Elapsed time(s)= 51.40 diff Utot= 0.000000000008 diff Force= 0.000000000055  
10 input_example/H2O.dat Elapsed time(s)= 36.75 diff Utot= 0.000000000000 diff Force= 0.000000000001  
11 input_example/HYb.dat Elapsed time(s)= 142.48 diff Utot= 0.000000000000 diff Force= 0.000000000000  
12 input_example/Methane.dat Elapsed time(s)= 21.23 diff Utot= 0.000000000000 diff Force= 0.000000000000  
13 input_example/Mo1-MnO.dat Elapsed time(s)= 141.55 diff Utot= 0.000000000014 diff Force= 0.000000000001  
14 input_example/Ndia2.dat Elapsed time(s)= 19.89 diff Utot= 0.000000000026 diff Force= 0.000000000011  
  
Total elapsed time (s) 1180.72700
```

Also, the results on several platforms can be found in [openm3.3/work/input_example](#)

Input file

```
Xterm
[ozaki@vtppcc01 work]$ more Methane.dat
#
# SCF calculation of a methane molecule by the LDA
# and the cluster method
#
# File Name
#
System.CurrentDirectory ./ # 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 H4,0-s1 H_TM
C C4,5-s1p1 C_TM_PCC
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) Value **behind** keyword
- (2) The order is **arbitrarily**.
- (3) Put # to the head of line for comment

Selected keywords:

The name of system

The number of atoms

Definition of species

Atomic coordinates

Output files

The amount of the output files is controlled by the keyword, ‘**level.of.fileout**’ (0,1,2). The typical default output files is as follows:

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.memory0	analysis for used memory
met.md	geometrical structures at every MD step
met.md2	geometrical structure of the final MD step
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_rst/	directory storing restart files

For more details, see the section ‘**Output files**’ of the manual.

Directories for basis sets and pseudopotentials

You will find the directories, **PAO** and **VPS** below **DFT_DATA**

```
X kterm
[ozaki@vtppcc01 DFT_DATA]$ pwd
/home/ozaki/openmx3.3/DFT_DATA
[ozaki@vtppcc01 DFT_DATA]$ ls
PAO VPS
[ozaki@vtppcc01 DFT_DATA]$ █
```

*.pao and *.vps files have to be stored in the directories **openmx3.3/DFT_DATA/PAO** , **openmx3.3/DFT_DATA/VPS** , respectively.

The directory **DFT_DATA** can be specified by the keyword as

```
DATA.PATH    ../DFT_DATA2006/    # default=../DFT_DATA/
```

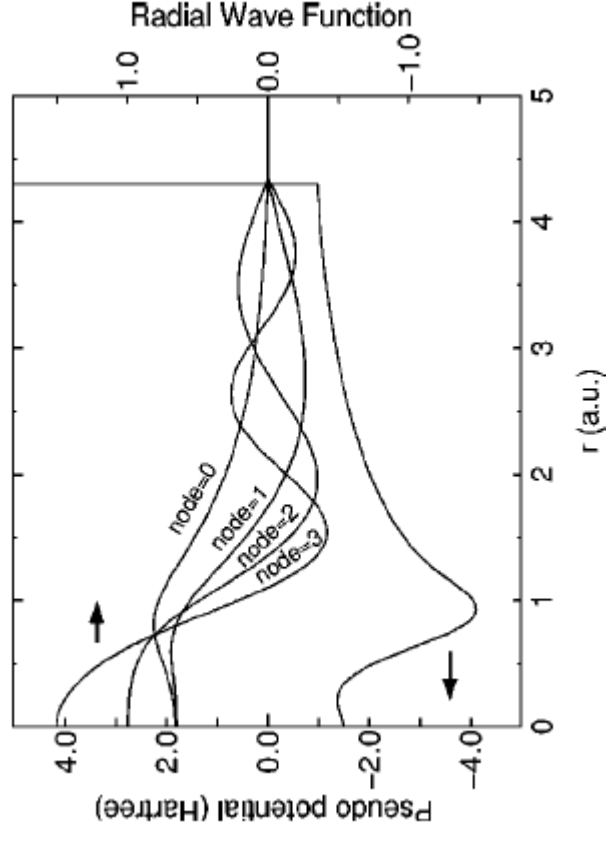
The keyword is useful to avoid using PAO files for pseudopotentials of the different versions

Basis sets No.1

The species of atoms can be specified by

```
Species.Number      2
<Definition.of.Atomic.Species
H   H4.0-s1         H_TM
C   C4.5-s1p1      C_TM_PCC
Definition.of.Atomic.Species>
```

where H4.0-s1 means that the first s-orbital of H4.0.pao is used, and C4.5-s1p1 means that the first s- and p-orbitals in C4.5.pao is used as basis functions.



Basis sets No.2

The **proper choice** of basis functions depends on **elements** and **chemical environment**. Although detailed analyses can be found in PRB 69, 195113 (2004) and JCP 121, 10879 (2004), **the following can be good starting points.**

Examples	H4.5-s2	Li8.0-s2	Ti5.5-s2p2d1	Fe5.5-s2p2d1
	B4.5-s2p1	Na9.0-s2	V5.5-s2p2d1	Co5.5-s2p2d1
	C4.5-s2p1	K9.0-s2	Cr5.5-s2p2d1	Ni5.5-s2p2d1
	N4.5-s2p2		Mn5.5-s2p2d1	Cu5.5-s2p2d1
	O4.5-s2p2d1			
	F4.5-s2p2d1			

Trends:

- (1) Elements located in the **right** side of the periodic table requires basis sets with **higher angular momentum**.
- (2) **Alkali** metals require a **long tail** of basis functions.
- (3) **3d-transition metals** are well described by **TM5.5-s2p2d1** in their **oxide**.

Pseudopotentials

The species of atoms can be specified by

```
Species.Number      2
<Definition.of.Atomic.Species
H   H4.0-s1         H_TM
C   C4.5-s1p1      C_TM_PCC
Definition.of.Atomic.Species>
```

H_TM and C_TM_PCC mean that H_TM.vps and C_TM_PCC.vps stored in the directory specified by DATA.PATH are used as pseudopotentials.

Functionals

Three kinds of functional are now available by the following keywords

```
scf .XcType          LDA          # LDA|LSDA-GA|LSDA-FW|GGA-PBE
scf .SpinPolarization off         # On|Off|MC
```

In case of LDA+U

```
scf .Hubbard.U      on          # On|Off , default=off
scf .Hubbard.Occupation dual     # onsite|full|dual, default=dual
```

On-site Us are specified by

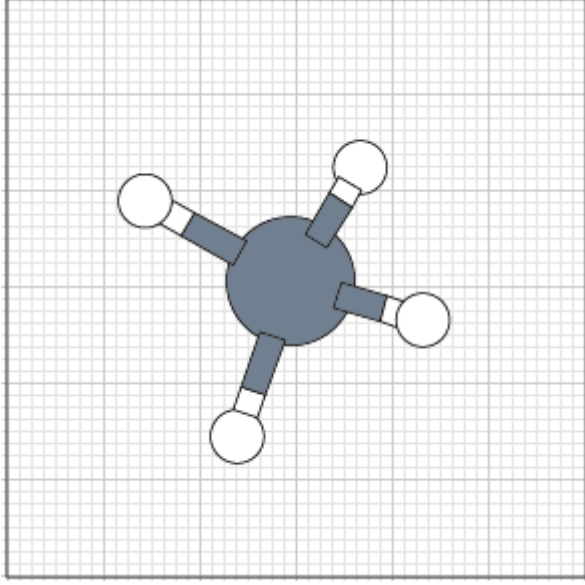
```
Species.Number      2
<Definition.of .Atomic.Species
Ni Ni5.5-s2p2d2f1   Ni_LDA
0  05.0-s2p2d1     0_LDA
Definition.of .Atomic.Species>

<Hubbard.U.values
Ni 1s 0.0 2s 0.0 1p 0.0 2p 0.0 1d 7.0 2d 0.0 1f 0.0
0  1s 0.0 2s 0.0 1p 0.0 2p 0.0 1d 0.0
Hubbard.U.values>
```

Cutoff energy for regular mesh No.1

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.

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

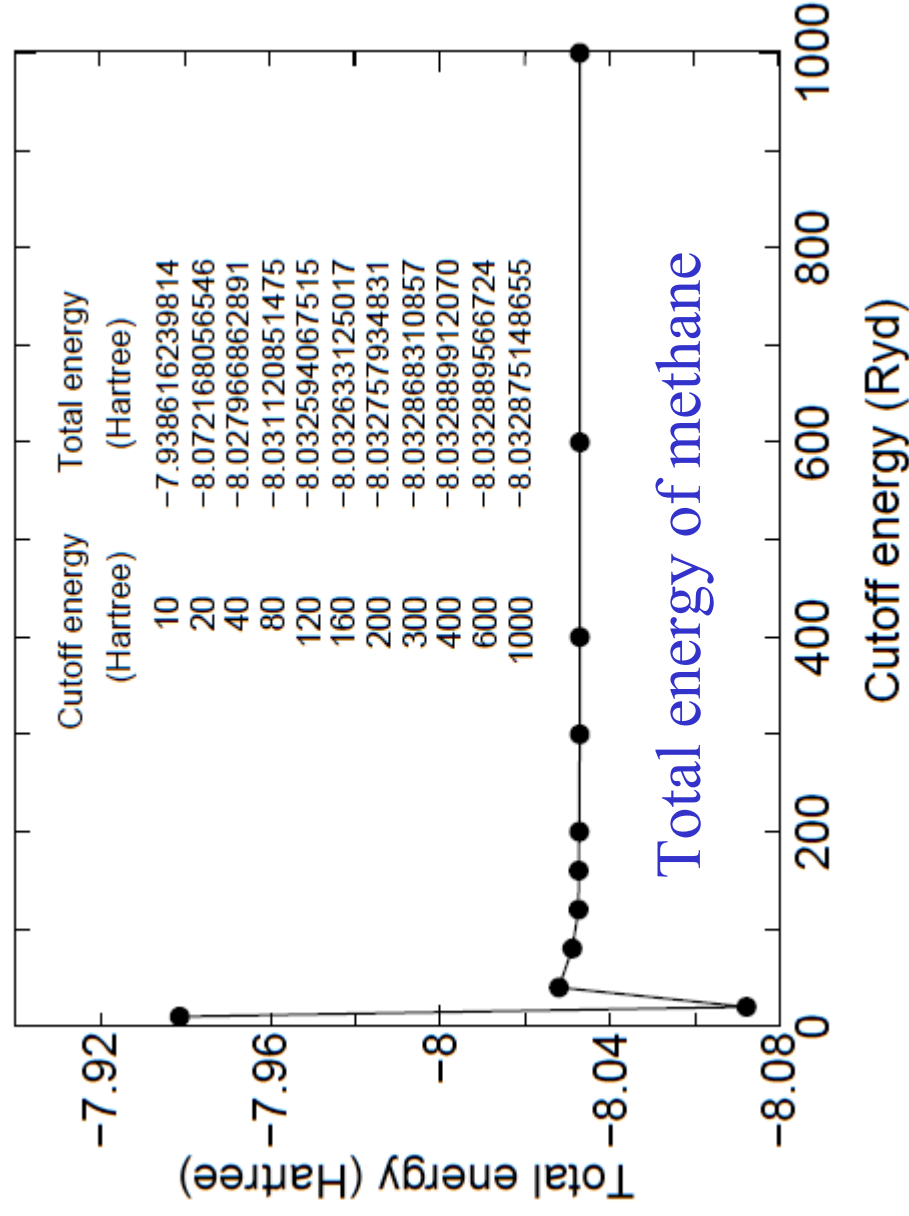
$$\mathbf{g} \mathbf{a}_1 = \frac{\mathbf{a}_1}{N_1}, \quad \mathbf{g} \mathbf{a}_2 = \frac{\mathbf{a}_2}{N_2}, \quad \mathbf{g} \mathbf{a}_3 = \frac{\mathbf{a}_3}{N_3},$$

$$\mathbf{g} \mathbf{b}_1 = 2\pi \frac{\mathbf{g} \mathbf{a}_2 \times \mathbf{g} \mathbf{a}_3}{\Delta V}, \quad \mathbf{g} \mathbf{b}_2 = 2\pi \frac{\mathbf{g} \mathbf{a}_3 \times \mathbf{g} \mathbf{a}_1}{\Delta V}, \quad \mathbf{g} \mathbf{b}_3 = 2\pi \frac{\mathbf{g} \mathbf{a}_1 \times \mathbf{g} \mathbf{a}_2}{\Delta V},$$

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

Cutoff energy for regular mesh No.2

The proper choice of the cutoff energy depends on system.
150-250 Ryd is a reasonable choice in most cases.



Getting SCF No.1

Five methods for getting SCF are available, the **RMM-DIISK** is the best choice among them in most cases.

```
scf.Mixing.Type      rmm-diisk      # Simple|Rmm-Diis|Gr-Pulay
                    # Kerker|Rmm-Diisk
```

The **RMM-DIISK** is controlled by the following keywords:

```
scf.Init.Mixing.Weight      0.0100      # default=0.30
scf.Min.Mixing.Weight       0.0010      # default=0.001
scf.Max.Mixing.Weight       0.1000      # default=0.40
scf.Kerker.factor           3.000      # default=1.00
scf.Mixing.History          20          # default=5
scf.Mixing.StartPulay       12          # default=6
scf.Mixing.EveryPulay       1           # default=6
scf.criterion                1.0e-10     # default=1.0e-6 (Hartree)
```

Getting SCF No.2

If you **cannot** obtain the SCF convergence, try the following prescriptions:

- **Decrease `scf.Min.Mixing.Weight`**. The large `scf.Min.Mixing.Weight` may lead to unfavorable charge sloshing
- **Use a rather larger value for `scf.Mixing.StartPulay`**. Before starting the Pulay type mixing, achieve a convergence at some level. An appropriate value may be 10 to 30 for `scf.Mixing.StartPulay`.
- **Use a rather larger value for `scf.ElectronicTemperature`** in case of metallic systems. When `scf.ElectronicTemperature` is small, numerical instabilities appear often.
- **Increase `scf.Mixing.History`**. `scf.Mixing.History=25-40` could lead to the SCF convergence in most cases.

Getting SCF No.3

Those parameters for 20 systems difficult to get the SCF convergence and the results can be found at

http://www.jaist.ac.jp/~t-ozaki/large_example.tar.gz

Geometry optimization No.1

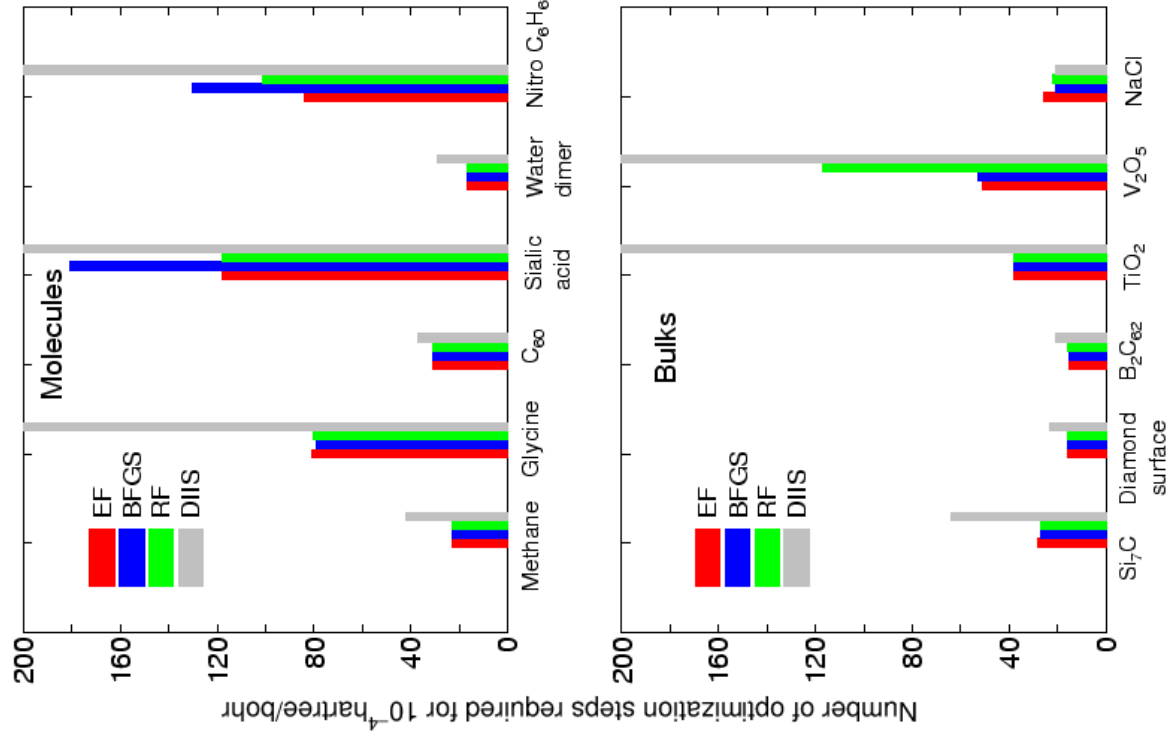
Five methods for geometry optimization are available, the **EF** is the best choice among them in most cases.

```
MD.Type          EF          # Opt|DIIS|BFGS|RF|EF
```

The **EF** is controlled by the following keywords:

```
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)
```

Geometry optimization No.2



The **EF** gives a faster convergence,
While the **RF** is also a good choice.

For your convenience, the input files and out files used in the calculations shown in the figure can be found in
"openmx3.3/work/geoopt_example".

Geometry optimization No.3

If you **cannot** obtain the optimized geometry, try the following prescriptions:

- **Increase MD.Opt.DIIS.History.** MD.Opt.DIIS.History=10-20 could lead to the SCF convergence in some cases (not all).
- **Use a rather larger value for MD.OptStart.DIIS.** Before starting the Pulay type mixing, achieve a convergence at some level. An appropriate value may be 10 to 30 for MD.OptStart.DIIS.
- **Use a rather larger value for MD.Opt.criterion.** There is a case that the maximum force does not decrease below 10^{-4} Hartree/bohr. Then, a compromise is to increase MD.Opt.criterion to 3.0×10^{-4} Hartree/bohr or more.

Initial spin moments

The initial spin moments can be controlled by the last **two columns** in the keyword **Atoms.SpeciesAndCoordinates**.

Example: MnO in the NaCl structure

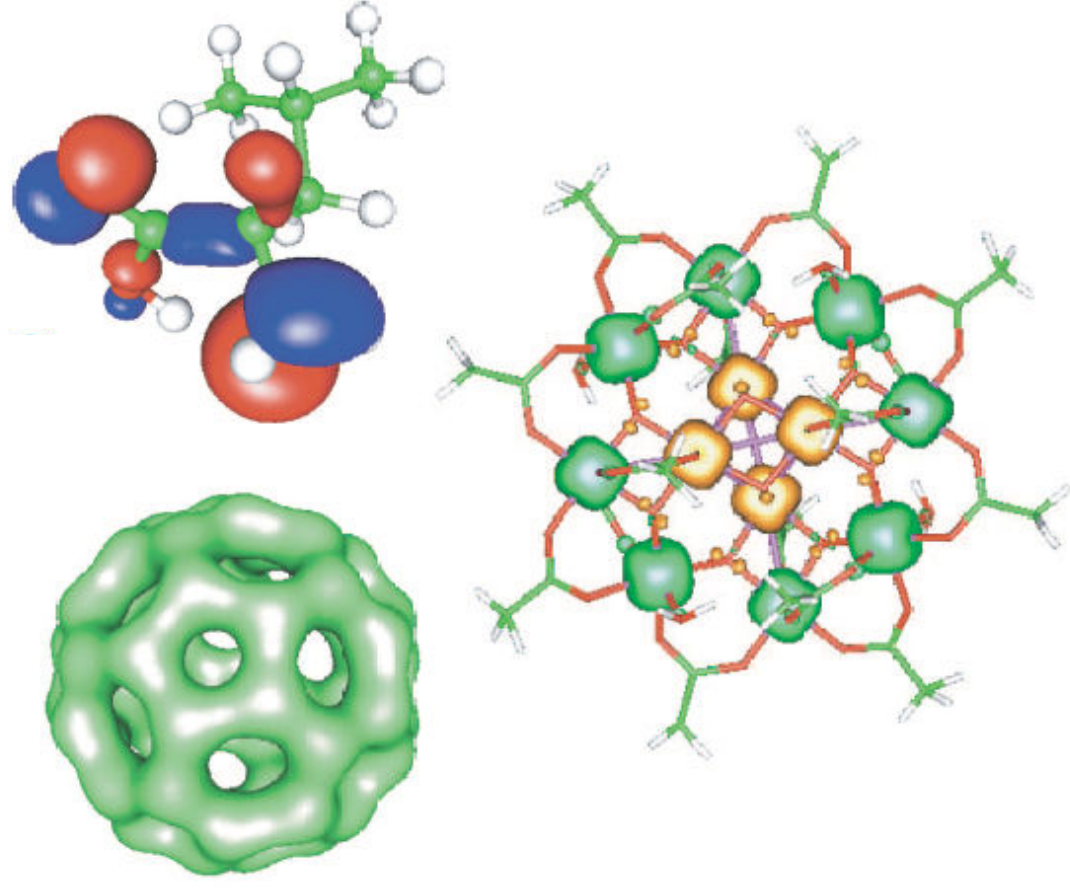
Ferromagnetic order

```
<Atoms.SpeciesAndCoordinates
1 Mn 0.000 0.000 0.000
2 Mn 4.1905 4.1905 0.000
3 O 4.1905 0.000 0.000
4 O 4.1905 4.1905 4.1905
Atoms.SpeciesAndCoordinates>
# Unit=AU
8.0 5.0
8.0 5.0
3.0 3.0
3.0 3.0
```

Antiferromagnetic order

```
<Atoms.SpeciesAndCoordinates
1 Mn 0.000 0.000 0.000
2 Mn 4.1905 4.1905 0.000
3 O 4.1905 0.000 0.000
4 O 4.1905 4.1905 4.1905
Atoms.SpeciesAndCoordinates>
# Unit=AU
8.0 5.0
5.0 8.0
3.0 3.0
3.0 3.0
```

Visualization of cube files



The generated **cube files** (*.cube) can be visualized by many software such as

XCrysDen
gOpenMol
Molkel
etc.

Other things

See the manual about

Density of states

Band dispersion

Charge analysis

Non-collinear DFT

Spin-orbit interaction

Linear scaling methods

Molecular dynamics

Applying electric field

Zeeman term

Electric polarization

etc.....

OpenMX Forum

If you **cannot** solve your problem, please post your problem to the OpenMX Forum.