

OpenMXハンズオン: 基本計算編

- クラスタ計算: gaussへのログイン
- ジョブの投げ方
- メタン分子のテスト計算
- 水素分子の結合エネルギーを計算する
- cifファイルを利用したSi結晶のバンド計算
- 構造最適化を実行する
- 仕事関数の計算
- STM像を計算する
- 内殻束縛エネルギーを計算する
- 非平衡グリーン関数法を用いて透過率を計算する

尾崎泰助

東京大学物性研究所

CCMSハンズオン OpenMX講習会 2022年2月22日

gaussへのログイン

各自のx-terminalからsshによりログイン

```
% ssh -YC guest01@gauss.issp.u-tokyo.ac.jp
```



アカウント名は各自に割り当てられたもの



gaussの基本情報

System

- * Login node : gauss
- * Computational nodes : gauss01 - gauss13
 - * CPU : Xeon E5-2680v4, 28 cores/node
 - * RAM : 64 GB/node
 - * Do not login directly the computational nodes.
Use the queuing system.
- * Sample of batch job script : /home/hands-on/openmx2022/sample.sh

Programs

- * Binary : /home/public/bin/
- * Source : /home/public/program/

まずジョブを投げてみる

(0) openmx3.9.tar.gzをcopyして解凍する

```
$ cp /home/hands-on/openmx2022/openmx3.9.tar.gz .
```

```
$ tar zxvf openmx3.9.tar.gz
```

(1) openmx3.9のworkディレクトリに移動する

```
$ cd openmx3.9/work
```

(2) batchジョブスクリプトをコピーする

```
$ cp /home/hands-on/openmx2022/sample.sh .
```

(3) ジョブを投げる

```
$ qsub sample.sh
```

sample.shの中身

```
[ozaki@gauss work]$ cat sample.sh
```

```
#!/bin/sh
```

```
#PBS -l nodes=1:ppn=10
```

```
#PBS -n
```

```
#
```

```
source ~/.bashrc
```

```
#
```

```
export OMP_NUM_THREADS=1
```

```
#
```

```
cd $PBS_O_WORKDIR
```

```
#
```

```
mpiexec -hostfile $PBS_NODEFILE /home/public/bin/openmx Methane.dat -nt  
$OMP_NUM_THREADS > ${PBS_JOBID}.out 2>&1
```

```
[ozaki@gauss work]$
```

ジョブを投げる際には適宜、入
力ファイル名を変更する



最初の計算

メタン分子のSCF計算を実行する。

```
[ozaki@gauss work]$ qsub sample.sh
```

計算が正常に終了すると様々なファイルが生成される。

```
[ozaki@gauss work]$ ls met.*  
met.cif      met.ene  met.md2  met.tden.cube  met.vhart.cube  
met.dden.cube  met.md   met.out  met.v0.cube   met.xyz  
[ozaki@gauss work]$
```

OpenMXにおけるinput file

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

- (1) Keywordの後に値を与える。
- (2) Keywordの順番は任意で良い。
- (3) コメント文は #を文頭に入れる。

```
#
# Definition of Atomic Species
#
```

いくつかのkeywords:

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

システム名：この名前で様々な
ファイルを保存

系に含まれる原子数

原子種の定義

```
#
# Atoms
#
```

原子数

```
Atoms.Number                5
Atoms.SpeciesAndCoordinates.Unit  Ang # Ang|AU
```

座標指定の単位

```
<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
```

原子の座標 (x, y, z)

アップ、ダウンスピン電子に対
する初期電子数

```
Atoms.SpeciesAndCoordinates>
```

基底関数の指定

基底関数は以下の様に指定する(水分子の場合)。

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

- O7.0はO7.0.paoに対応する。
- -s2p2d1は動径関数として、s軌道を二つ、p軌道を2つ、d軌道を1つ選択することを意味する。
- この時、酸素Oに対して $2 \times 1 + 2 \times 3 + 1 \times 5 = 13$ 基底が割り当てられることになる。

O7.0.paoやO_PBE19.vpsの保存先は以下のkeywordで指定する。

```
DATA.PATH  ../DFT_DATA19  # default=../DFT_DATA19
```

Default値は../DFT_DATA19である。

出力ファイル

例えばMethane.datの計算が正常終了した場合、11個のファイルと一つのディレクトリが生成される。

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

met.outの中身

SCFの履歴

```
*****
*****
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= -3.891668816209
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
```

KS固有値

```
*****
*****
Eigenvalues (Hartree) for SCF KS-eq.
*****
*****
Chemical Potential (Hartree) = 0.000000000000
Number of States = 8.000000000000
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
```

全エネルギー

```
*****
Total energy (Hartree) at MD = 1
*****
Uele.      -3.889285101063
Ukin.      5.533754016241
UHO.      -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
```

マリケン電荷

```
*****
*****
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 Mull: up = 4.00000 down = 4.00000
total= 8.00000 ideal(neutral)= 8.00000
```

標準出力

```
 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
```

標準出力を保存したmet.stdを確認することで、SCFの途中経過を確認することができる。

```
***** 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とSCFのステップ数

各原子のマリケン電荷

Kohn-Sham固有値エネルギーと前回のSCFステップでの値との差分

```
***** 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
```

前回のSCFステップでの電荷密度との差分

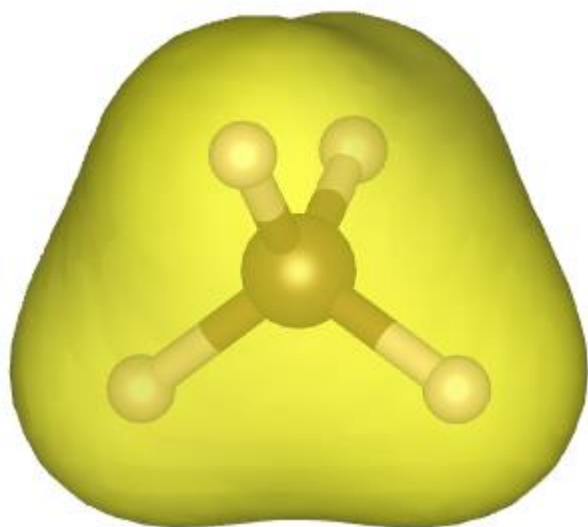
SCFの収束条件

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

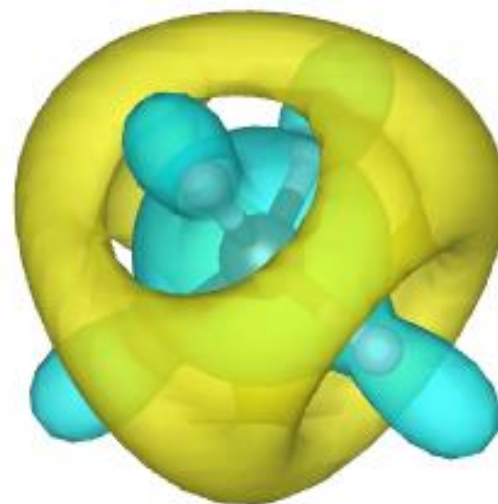
CubeファイルをVESTAで可視化する

メタン分子の計算後に電子密度やクーロンポテンシャルのcubeファイルが生成する。VESTAを用いて可視化できる。

met.tden.cube



met.vhart.cube



VESTAのwebsite <http://jp-minerals.org/vesta/jp/>

水素分子の結合エネルギーを計算する

水素分子の結合エネルギーを計算する

(1) openmx3.9のworkディレクトリにファイルをコピーする

```
$ pwd
```

```
$ cp /home/hands-on/openmx2022/H.dat .
```

```
$ cp /home/hands-on/openmx2022/H2.dat .
```

(2) sample.shにおいて入力ファイルを順次、指定してジョブを投げる

h.outとh2.outから全エネルギーを取得

h.outとh2.outを開いて、全エネルギーを取得する。以下の値が得られる。

水素原子 Utot. -0.499351255995

水素分子 Utot. -1.165981886187

したがって結合エネルギーは以下のように計算できる。

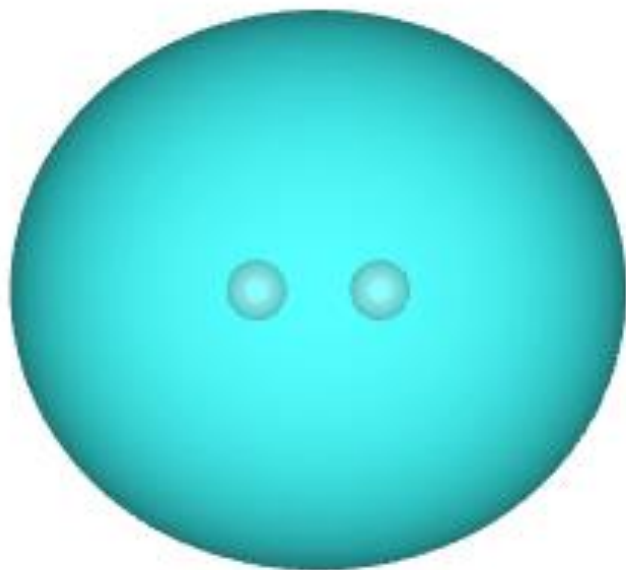
$$\begin{aligned}\text{結合エネルギー} &= 2 \text{ H} - \text{H}_2 \\ &= 2 \times (-0.49935) - (-1.16598) \\ &= 0.1673 \text{ (Hartree)} \\ &= 4.55 \text{ (eV)}\end{aligned}$$

実験値 4.75 (eV)

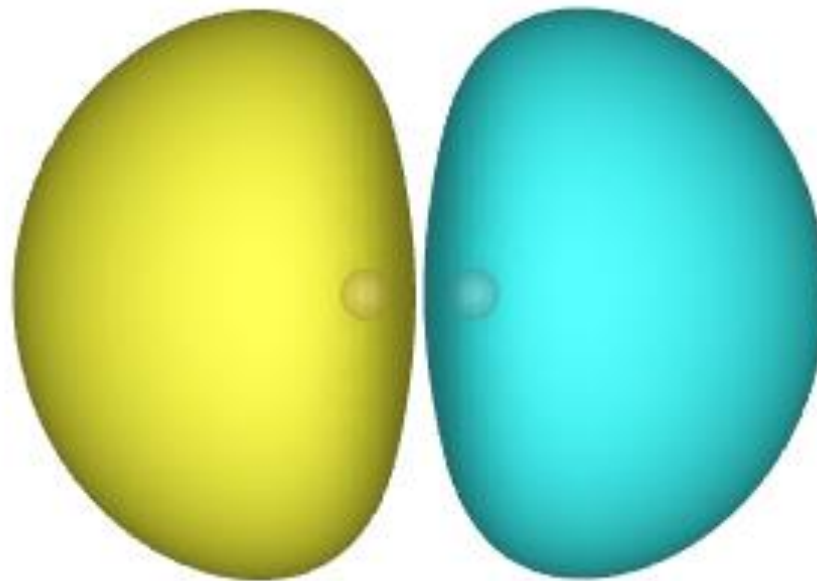
CubeファイルをVESTAで可視化する

水素分子の計算後にHOMOとLUMOに対するcubeファイルが生成する。
VESTAを用いて可視化できる。

h2.homo0_0_r.cube



h2.lumo0_0_r.cube



VESTAのwebsite <http://jp-minerals.org/vesta/jp/>

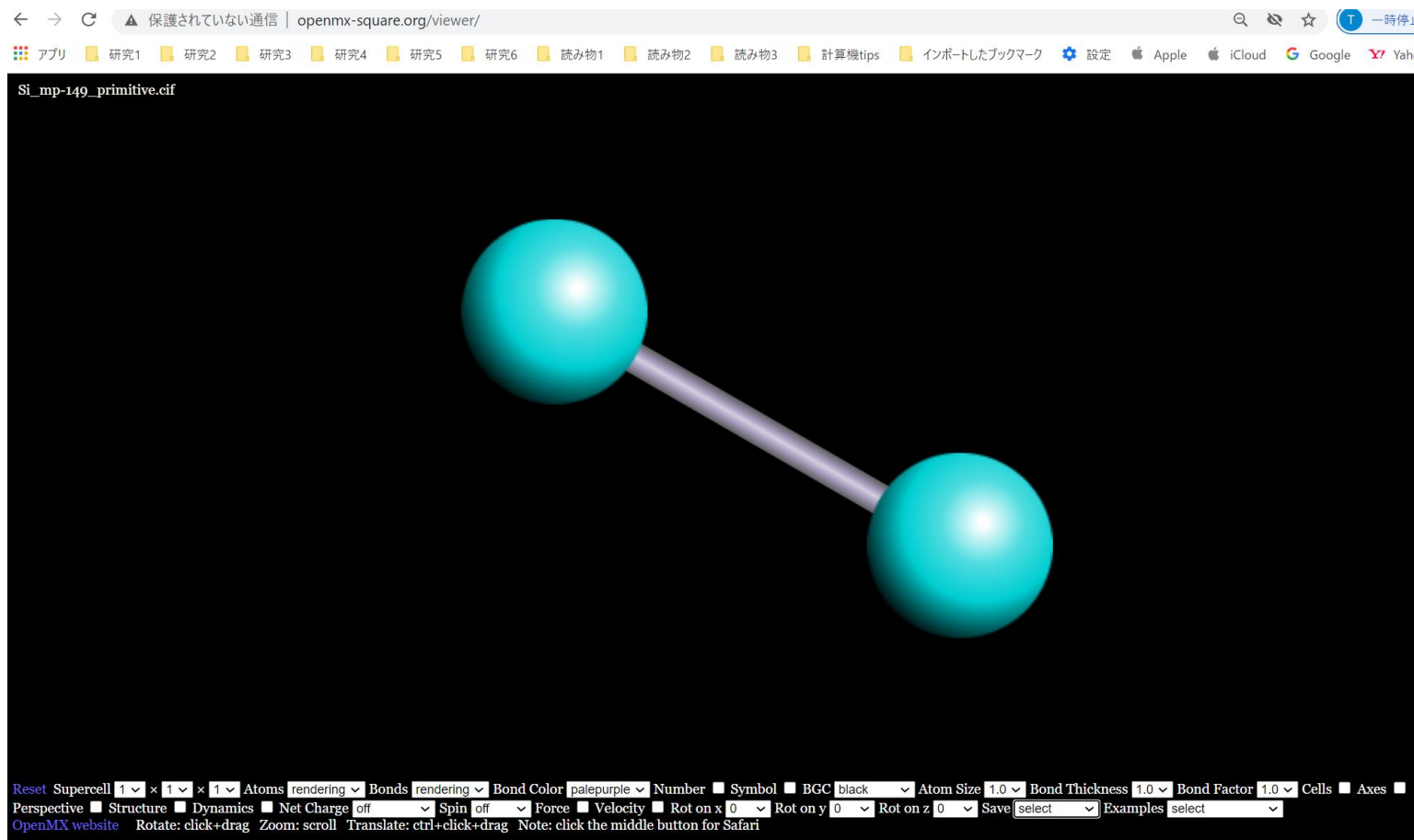
cifファイルを利用したSi結晶のバンド計算

cifファイルがある場合 #1

Si_mp-149_primitive.cif (を以下よりダウンロード)

http://www.openmx-square.org/hands-on2022/Si_mp-149_primitive.cif

をOpenMX Viewerにdrag & dropする。 <http://www.openmx-square.org/viewer/>



The screenshot displays the OpenMX Viewer web interface. The browser address bar shows the URL <http://www.openmx-square.org/viewer/>. The main content area shows a 3D ball-and-stick model of a diatomic molecule, with two cyan spheres representing atoms connected by a grey bond. The interface includes a top navigation bar with various tabs and a bottom control panel with numerous settings for rendering and simulation. The title of the viewer is "Si_mp-149_primitive.cif".

cifファイルがある場合 #2

(1) OMX(frac)を選ぶ

Atom Size 1.0 ▾ Bond Thickness 1.0 ▾
n z 0 ▾ Save select ▾ Examples se

(2) abc.datがsaveされる

Reset Super cell 1 ▾ × 1 ▾ × 1 ▾ Atoms
Perspective Structure Dynamics N
OpenMX website Rotate: click+drag Zoom

abc.dat

abc.datはOpenMXの入力ファイルとなっている。

```
#
# This was generated by OpenMX Viewer
System.CurrentDirectory      ./
System.Name                  abc
level.of.stdout              1
level.of.fileout             1

Species.Number               1
<Definition.of.Atomic.Species
  Si  Si7.0-s2p2d1  Si_PBE19
Definition.of.Atomic.Species>

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>

scf.XcType                   GGA-PBE
scf.SpinPolarization          off
scf.ElectronicTemperature     300.0
scf.energycutoff              220.0
scf.maxIter                   100
scf.EigenvalueSolver          band
scf.Kgrid                      6 6 6
```

cifファイルがある場合 #3

Seek-pathにcifファイルをuploadすれば、バンド図を計算するための経路が得られる。

Seek-path: the k-path finder and visualizer

Warning:

- I tried my best, but I wasn't able to load your file in format 'qeinp-qetools'...

▶ What Seek-path does

▶ Seek-path definitions and advantages

Upload your structure

Upload a crystal structure:

ファイルを選択

Select here the file format:

CIF File (.cif) [parser: ase]

By continuing, you agree with the terms of use of this service.

Calculate my structure

Otherwise, pick an example

Select here an extended Bravais Symbol:

aP2 [with inversion]

A simple explanation of the extended Bravais symbols.

Calculate this example

Reciprocal space and Brillouin-zone information

Reciprocal cell vectors (1/Å)

b	x	y	z
b₁	-1.1489299363	1.1489299363	1.1489299363
b₂	1.1489299363	-1.1489299363	1.1489299363
b₃	1.1489299363	1.1489299363	-1.1489299363

Suggested path

Γ—X—U|K—Γ—L—W—X

High-symmetry points (scaled units)

Label	k₁	k₂	k₃
Γ	0.0000000000	0.0000000000	0.0000000000
K	0.3750000000	0.3750000000	0.7500000000
L	0.5000000000	0.5000000000	0.5000000000
U	0.6250000000	0.2500000000	0.6250000000
W	0.5000000000	0.2500000000	0.7500000000
W ₂	0.7500000000	0.2500000000	0.5000000000
X	0.5000000000	0.0000000000	0.5000000000

<https://www.materialscloud.org/work/tools/seekpath>

cifファイルがある場合 #4

abc.datをSidia.datと名前を変更。またDOSとバンド図を計算するためのkeywordsを付け加える。

DOSを計算するためのkeywords:

```
Dos.fileout      on
Dos.Erange      -15.0 10.0
Dos.Kgrid       12 12 12
```

状態密度の計算方法に関しては

http://www.openmx-square.org/openmx_man3.9jp/node70.html
を参照のこと

バンド図を計算するためのkeywords:

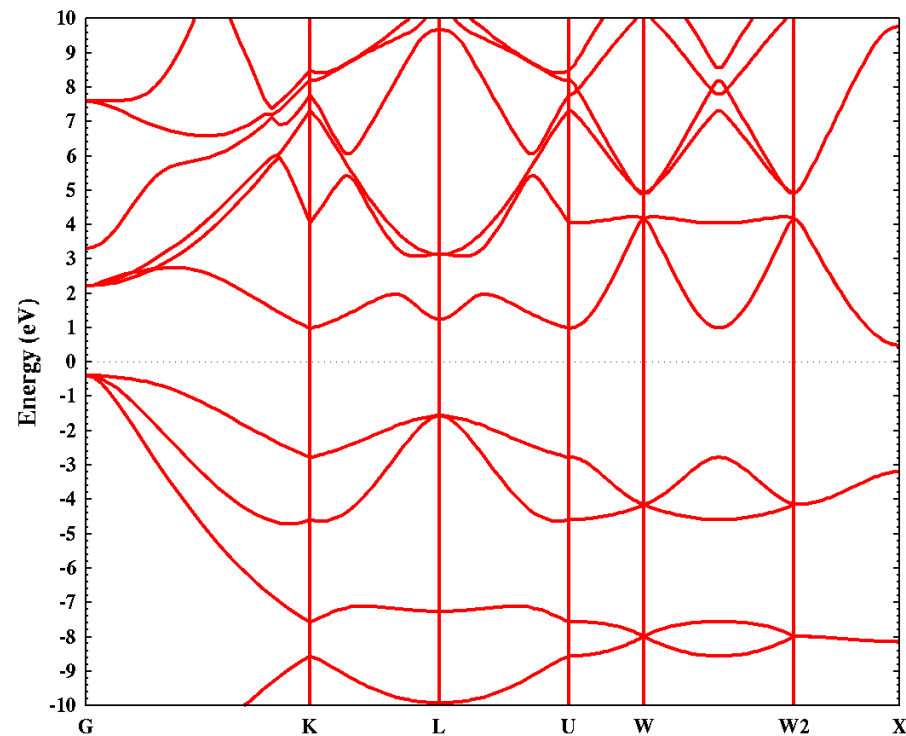
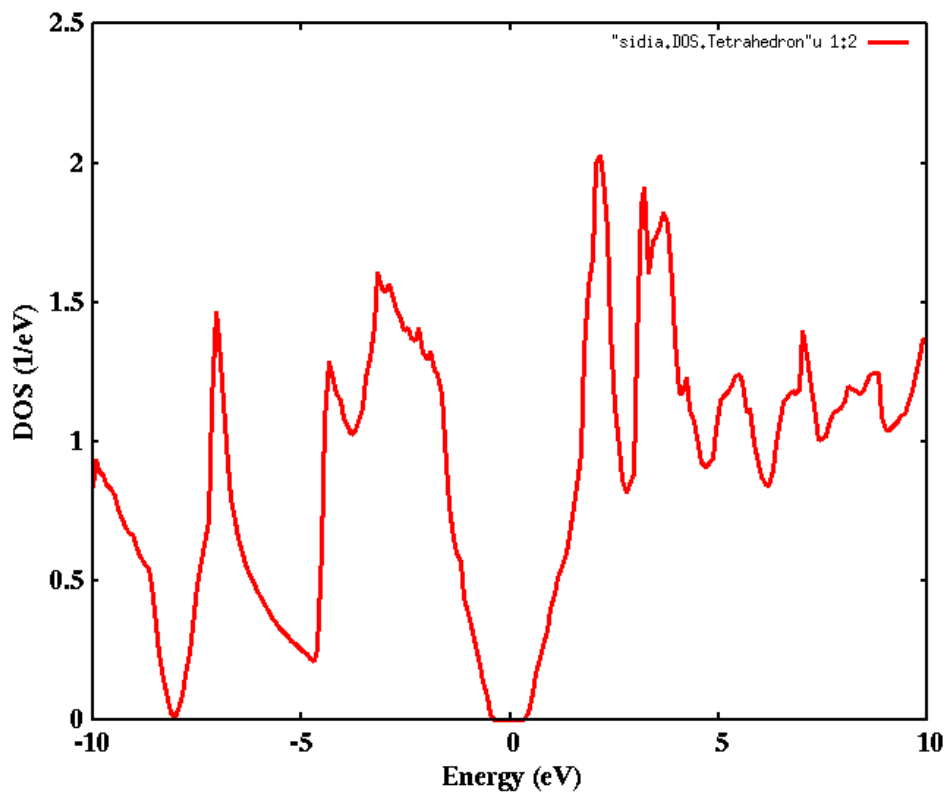
```
Band.dispersion  on      # on|off, default=off
Band.Nkpath      6
<Band.kpath
30 0.0000000000 0.0000000000 0.0000000000 0.3750000000 0.3750000000 0.7500000000 G K
30 0.3750000000 0.3750000000 0.7500000000 0.5000000000 0.5000000000 0.5000000000 K L
30 0.5000000000 0.5000000000 0.5000000000 0.6250000000 0.2500000000 0.6250000000 L U
30 0.6250000000 0.2500000000 0.6250000000 0.5000000000 0.2500000000 0.7500000000 U W
30 0.5000000000 0.2500000000 0.7500000000 0.7500000000 0.2500000000 0.5000000000 W W2
30 0.7500000000 0.2500000000 0.5000000000 0.5000000000 0.0000000000 0.5000000000 W2 X
Band.kpath>
```

バンド図の計算方法に関しては

http://www.openmx-square.org/openmx_man3.9jp/node68.html
を参照のこと

cifファイルがある場合 #5

abc.datをSidia.datと名前を変更。またDOSとバンド図を計算するためのkeywordsを付け加えて、計算を実行。下記の図が得られる。



状態密度の計算方法に関しては

http://www.openmx-square.org/openmx_man3.9jp/node70.html

を参照のこと

バンド図の計算方法に関しては

http://www.openmx-square.org/openmx_man3.9jp/node68.html

を参照のこと

Siのバンド計算の入力ファイル #1

System.CurrentDirectory ./

Keywordは大文字、小文字を区別しない。
Keywordの順番に決まらない。

このキーワードにより、出力ファイルの出力ディレクトリを指定。デフォルト値は「./」。

System.Name sidia

このキーワードで出力ファイルのファイル名を指定。

level.of.stdout 1

標準出力への出力情報のレベルを制御。「level.of.stdout=0」を指定した場合、最小限の情報が出力。「level.of.stdout=1」の場合、最小限の出力に加えて追加の情報が出力。「level.of.stdout=2」は開発者向けのオプション。デフォルト値は1。

level.of.fileout 1

このキーワードにより、出力ファイルへの出力情報のレベルを制御。「level.of.fileout=0」を指定した場合、最小限の情報が出力（Gaussian cubeファイルおよびグリッドファイルの出力無し）。「level.of.fileout=1」は標準的な出力レベル。「level.of.fileout=2」の場合、標準の出力に加えて追加の情報が出力。デフォルト値は1。

data.path ../

このキーワードにより、VPSおよびPAOディレクトリへのパスを指定。相対パス、絶対パスのどちらでも指定可能。

Siのバンド計算の入力ファイル #2

Species.Number 1

このキーワードにより、系に含まれる原子種の数指定。

<Definition.of.Atomic.Species

Si Si7.0-s2p2d1 Si_PBE19

Definition.of.Atomic.Species>

擬原子基底関数と擬ポテンシャルの両方のファイル名を明記することにより、原子種を指定。これらのファイルはそれぞれ「DFT_DATA19/PAO」および「DFT_DATA19/VPS」ディレクトリにある。

- Si7.0はSi7.0.paoに対応する。
- -s2p2d1は動径関数として、s軌道を二つ、p軌道を2つ、d軌道を1つ選択することを意味する。
- この時、ケイ素Siに対して $2 \times 1 + 2 \times 3 + 1 \times 5 = 13$ 基底が割り当てられることになる。

Si7.0.paoやSi_PBE19.vpsの保存先は以下のkeywordで指定。

```
DATA.PATH ../DFT_DATA19 # default=../DFT_DATA19
```

Default値は ../DFT_DATA19

Siのバンド計算の入力ファイル #3

Atoms.Number

2

系の合計原子数を「Atoms.Number」キーワードで指定。

Atoms.SpeciesAndCoordinates.Unit FRAC

原子座標の単位を指定するキーワード。

オングストローム → Ang

原子単位 → AU

規格化座標 → FRAC

規格化座標の場合には、「Atoms.UnitVectors」のキーワードで与えられた**a**、**b**、**c**ベクトルを基底として座標を指定。その際には0.0～1.0の範囲で座標が設定可能で、この範囲を超える座標は入力ファイルの読み込み後に自動的に調整される。

Siのバンド計算の入力ファイル #4

原子座標および各スピン毎の電子数をしてするキーワード

```
<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>
```

1列目: 原子を特定する連番

2列目: 「Definition.of.Atomic.Species」の1列目で定義した原子種を指定

3~5列目: それぞれx、y、z座標を指定

「Atoms.SpeciesAndCoordinates.Unit」キーワードで「FRAC」を指定している場合は3~5列目はそれぞれa、b、c軸の規格化座標になり、値は0.0~1.0の範囲で指定。

6,7列目:

各原子のアップスピンとダウンスピン状態の初期電子数を設定。原子の価電子数はhttp://www.openmx-square.org/openmx_man3.9jp/node27.html

強磁性状態や反強磁性状態などの計算ではアップスピンとダウンスピンの初期電子数を変えて、初期スピン配置を制御する。

Siのバンド計算の入力ファイル #5

Atoms.UnitVectors.Unit Ang

単位胞(ユニットセル)におけるベクトルの単位を指定するキーワード。
オングストロームの場合は「Ang」、原子単位の場合は「AU」を指定。

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

単位胞のベクトルa、b、cを指定するキーワード。第1～3行はそれぞれ
単位胞のベクトル**a**、**b**、**c**に対応。

Siのバンド計算の入力ファイル #6

scf.XcType GGA-PBE

交換相関汎関数を指定するキーワード。

Ceperley-Alder LDA ⇒ LDA、
Ceperley-AlderLSDA ⇒ LSDA-CA
Perdew-Wang LSDA ⇒ LSDA-PW
GGA by PBE ⇒ GGA-PBE

scf.SpinPolarization off

非スピン分極あるいはスピン分極を指定するキーワード。スピン分極計算を行う場合は「ON」を指定、非スピン分極の計算を行う場合は「OFF」。「scf.XcType」キーワードで「LDA」を使用する場合は「scf.SpinPolarization」を「OFF」に設定。

scf.ElectronicTemperature 300.0

電子温度 (K) を設定するキーワード。デフォルトでは300 (K) に設定。

scf.energycutoff 220.0

積分グリッド間隔を決定するカットオフエネルギーを指定。
デフォルト値は150 (Ryd) 。

Siのバンド計算の入力ファイル #7

scf.maxIter 100

SCFの最大反復回数を指定する。SCF反復ループは、収束条件が満たされない場合でもこのキーワードで指定した回数で終了する。デフォルト値は40。

scf.EigenvalueSolver band

固有値問題の計算手法を指定する。O(N)分割統治法は「DC」、局在自然軌道によるO(N)分割統治法は「DC-LNO」、O(N)クリロフ部分空間法は「Krylov」、数値厳密な低次スケーリング法は「ON2」、クラスタ計算は「Cluster」、バンド計算は「Band」を指定。

scf.Kgrid 6 6 6

「scf.EigenvalueSolver」キーワードにより「Band」を指定した場合、k空間の第1ブリルアンゾーンを離散化するための格子の数 (n_1, n_2, n_3) を「scf.Kgrid」キーワードで指定。k空間の逆格子ベクトル \mathbf{a} 、 \mathbf{b} 、 \mathbf{c} を離散化するために「n1 n2 n3」のように指定する。

Siのバンド計算の入力ファイル #8

scf.Mixing.Type **rmm-diisk**

本キーワードによって、SCF計算の次の反復ステップに入力される電子密度(もしくは密度行列)を生成するための電子密度混合法を指定。単純混合法 (Simple)、GR-Pulay (Guaranteed-Reduction Pulay)、RMM-DIIS、Kerker、RMM-DIISK、RMM-DIISV、RMM-DIISH のいずれかを選択。

scf.Init.Mixing.Weight **0.05**

初期の混合比を指定するキーワード。有効な範囲は $0 < \text{scf.Init.Mixing.Weight} < 1$ 。デフォルト値は0.3。

scf.Min.Mixing.Weight **0.01**

単純混合法およびKerker混合法における混合比の下限を指定するキーワード。デフォルト値は0.001。

scf.Max.Mixing.Weight **0.30**

単純混合法およびKerker混合法における混合比の上限を指定するキーワード。デフォルト値は0.001。

scf.Mixing.History **25**

GR-Pulay、RMM-DIIS、RMM-DIISK、RMM-DIISV、RMM-DIISHでは、SCFの次の反復ステップにおける入力電子密度 (ハミルトニアン) を、過去のSCF反復の電子密度 (ハミルトニアン) に基づき推定します。本キーワードにより、この推定に使用する過去のSCF反復ステップ数を指定します。例えば、「scf.Mixing.History」を3に設定した場合、6回目のSCF反復では過去の第5、4、3ステップの電子密度 (ハミルトニアン) が考慮されます。30程度が適切な選択値です。

scf.Mixing.StartPulay **15**

GR-Pulay、RMM-DIIS、Kerker、RMM-DIISK、RMM-DIISV、RMM-DIISHを開始するSCFステップを指定。これらの方法を開始するまでのSCFステップでは単純混合法あるいはKerker混合法が使用。デフォルトでは6。

scf.criterion **1.0e-7**

SCF計算の収束条件 (Hartree単位)

Siのバンド計算の入力ファイル #9

DOSを計算するためのkeywords:

```
Dos.fileout      on
Dos.Erange      -15.0 10.0
Dos.Kgrid       12 12 12
```

状態密度の計算方法に関しては

http://www.openmx-square.org/openmx_man3.9jp/node70.html
を参照のこと

バンド図を計算するためのkeywords:

```
Band.dispersion  on      # on|off, default=off
```

```
Band.Nkpath     6
```

```
<Band.kpath
```

```
30 0.0000000000 0.0000000000 0.0000000000 0.3750000000 0.3750000000 0.7500000000 G K
```

```
30 0.3750000000 0.3750000000 0.7500000000 0.5000000000 0.5000000000 0.5000000000 K L
```

```
30 0.5000000000 0.5000000000 0.5000000000 0.6250000000 0.2500000000 0.6250000000 L U
```

```
30 0.6250000000 0.2500000000 0.6250000000 0.5000000000 0.2500000000 0.7500000000 U W
```

```
30 0.5000000000 0.2500000000 0.7500000000 0.7500000000 0.2500000000 0.5000000000 W W2
```

```
30 0.7500000000 0.2500000000 0.5000000000 0.5000000000 0.0000000000 0.5000000000 W2 X
```

```
Band.kpath>
```

バンド図の計算方法に関しては

http://www.openmx-square.org/openmx_man3.9jp/node68.html
を参照のこと

構造最適化を実行する

メタン分子の構造最適化

(1) work中の「Methane2.dat」を用いて構造最適化を実施。ただし入力ファイル中で次のように変更する。

```
<Definition.of.Atomic.Species
H H5.0-s2 H_PBE19
C C5.0-s2p2 C_PBE19
Definition.of.Atomic.Species>
```

ここに注意

初期構造は以下の通り。

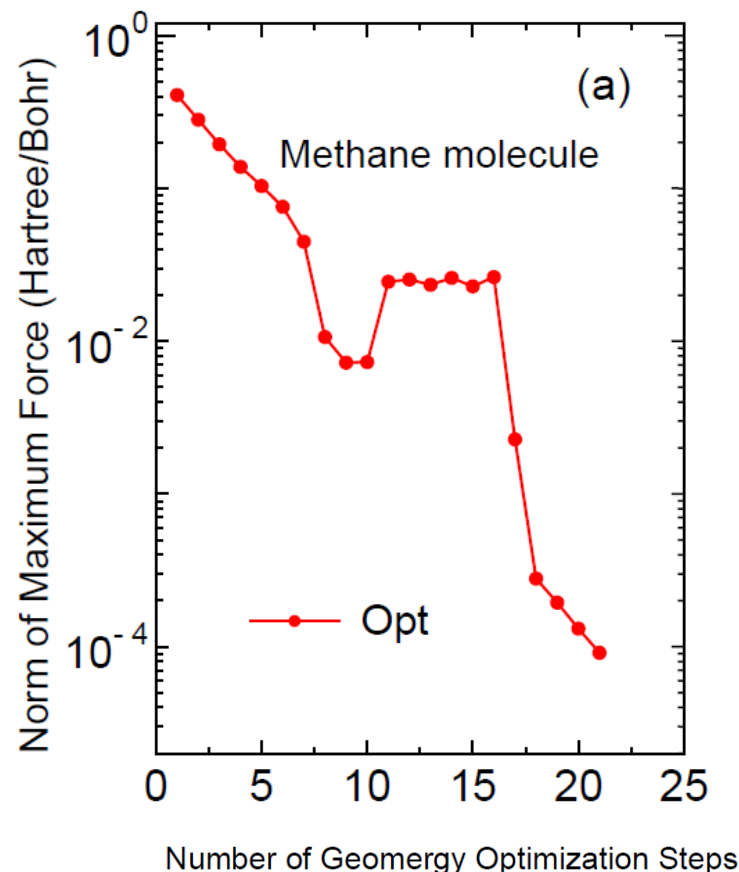
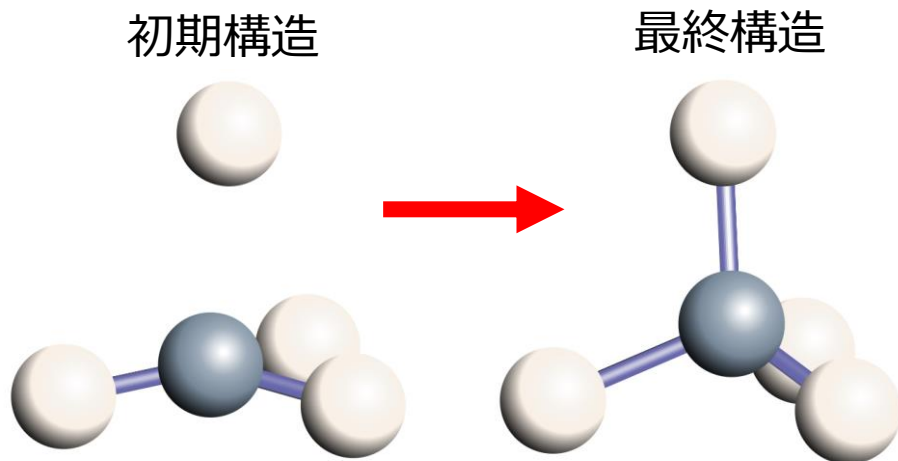
```
<Atoms.SpeciesAndCoordinates
```

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>
```

(3) met2.mdをOpenMX Viewerにdrag&dropすると最適化過程が可視化できる。

(2) met2.outのHistory of geometry optimizationをプロットすると以下のようなになる。



メタン分子のマリケン電荷

構造最適化後のマリケン電荷

```
Total spin moment (muB) 0.000000000
```

		Up spin	Down spin	Sum	Diff
1	C	2.039552637	2.039552637	4.079105274	0.000000000
2	H	0.490111840	0.490111840	0.980223681	0.000000000
3	H	0.490112112	0.490112112	0.980224224	0.000000000
4	H	0.490112144	0.490112144	0.980224288	0.000000000
5	H	0.490111267	0.490111267	0.980222533	0.000000000

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

軌道分解されたマリケン電荷

```
Decomposed Mulliken populations
```

1	C	Up spin	Down spin	Sum	Diff
s	multiple 0	0.534415255	0.534415255	1.068830511	0.000000000
sum over m		0.534415255	0.534415255	1.068830511	0.000000000
s	1	-0.007755050	-0.007755050	-0.015510100	0.000000000
sum over m		-0.007755050	-0.007755050	-0.015510100	0.000000000
sum over m+mul		0.526660205	0.526660205	1.053320411	0.000000000
px	0	0.515913947	0.515913947	1.031827894	0.000000000
py	0	0.515902481	0.515902481	1.031804963	0.000000000
pz	0	0.515915494	0.515915494	1.031830987	0.000000000
sum over m		1.547731922	1.547731922	3.095463844	0.000000000
px	1	-0.011613481	-0.011613481	-0.023226962	0.000000000
py	1	-0.011612251	-0.011612251	-0.023224502	0.000000000
pz	1	-0.011613758	-0.011613758	-0.023227516	0.000000000
sum over m		-0.034839490	-0.034839490	-0.069678980	0.000000000
sum over m+mul		1.512892432	1.512892432	3.025784864	0.000000000

OpenMX Viewerにmet2.mdをdrag&dropし、 Net Charge Mulliken とすれば可視化できる。

セル可変の構造最適化

(1) work/cellopt_example/Cdia-RFC5.datを自分の"work"ディレクトリにコピー。

(2) 計算を実行。

```
$ mpirun -np 4 ./openmx Cdia-RFC5.dat | tee Cdia-RFC5.std
```

(3) Cdia-RFC5.out中に以下の最適化過程が確認できる。

```
*****
*****
History of cell optimization
*****
*****
```

MD_iter	SD_scaling	Maximum force (Hartree/Bohr)	Maximum step (Ang)	Utot (Hartree)	Enpy (Hartree)	Volume (Ang ³)
1	1.25981732	0.15559460	0.10372973	-11.59621750	-11.59621750	8.82188800
2	1.25981732	0.10622032	0.07081354	-11.63880400	-11.63880400	9.84895362
3	1.25981732	0.08212391	0.05474928	-11.65321943	-11.65321943	10.33679488
4	1.25981732	0.06383589	0.04255726	-11.66078662	-11.66078662	10.62476618
5	3.14954331	0.05055414	0.08425689	-11.66535191	-11.66535191	10.80685020
6	3.14954331	0.02507162	0.04178604	-11.67169531	-11.67169531	11.10826026
7	3.14954331	0.01442607	0.01495779	-11.67387928	-11.67387928	11.17745805
8	3.14954331	0.00342018	0.01228241	-11.67490507	-11.67490507	11.21347036
9	3.14954331	0.00322947	0.01119394	-11.67523944	-11.67523944	11.26623064
10	3.14954331	0.00116703	0.00116984	-11.67537384	-11.67537384	11.33801088
11	3.14954331	0.00102668	0.00159452	-11.67538086	-11.67538086	11.35519285
12	3.14954331	0.00037750	0.00108136	-11.67538868	-11.67538868	11.38813287
13	3.14954331	0.00002292	0.00000000	-11.67538984	-11.67538984	11.39528546

(4) 最適化格子定数は3.572 Å (実験値は3.567 Å)であることが分かる。

以下webも参照のこと。

http://www.openmx-square.org/openmx_man3.9jp/node54.html

制約条件付きセル可変構造最適化

MD.TypeにRFC5を用いた場合、

MD.Type	RFC5
---------	------

MD.Fixed.Cell.Vectorsでセルベクトルに対して任意の制約条件を課すことが可能である (Ver. 3.9.9において)。

```
<MD.Fixed.Cell.Vectors
  0 0 1
  0 0 0
  0 0 0
MD.Fixed.Cell.Vectors>
```

ここでそれぞれの行はa,b,c軸のx,y,z座標に対応し、1は固定、0は最適化を意味する。

マニュアルの記述と異なることに注意。

仕事関数の計算

仕事関数の計算

真空領域でのポテンシャルと化学ポテンシャルの差から仕事関数を求める。

以下webに従い、Alの仕事関数を計算する。

http://www.openmx-square.org/openmx_man3.9jp/node153.html

work中にあるAl111_WorkFunc_2E.datを用いる。
いくつかの仕事関数の計算結果を以下にまとめる。

	OpenMX	Expt.
Al(111)	4.19	4.26 ± 0.03 [129]
Cu(111)	4.74	4.94 ± 0.03 [130]
Ag(111)	4.51	4.46 ± 0.02 [131]
Au(111)	5.33	5.26 ± 0.04 [132]
Pd(111)	5.40	5.55 ± 0.01 [133]

STM像を計算する

Tersoff-Hamann法によるSTM像の計算

Tersoff-Hamann法によって走査型トンネル顕微鏡 (STM) 像を計算する。この方法は、化学ポテンシャルから計ったエネルギーウィンド内の部分電荷密度の計算と等価である。以下webに従い計算を実施すれば、以下の様な像が得られる。

http://www.openmx-square.org/openmx_man3.9jp/node163.html

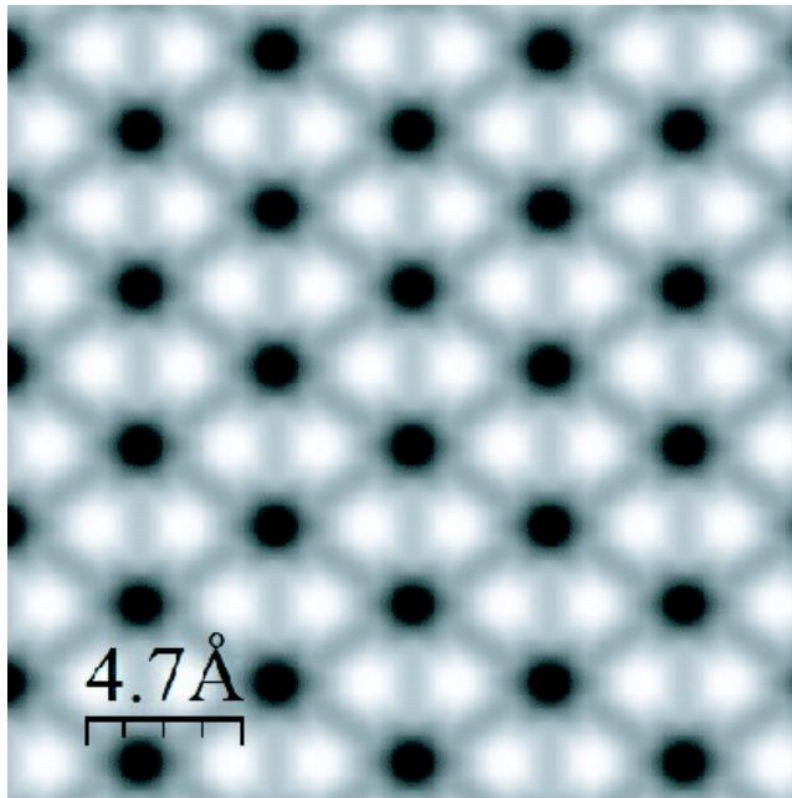


Figure 59: Tersoff-Hamann近似によるグラフェン層のSTM像。計算では「partial.charge.energy.window」に2eVに設定。入力ファイルはディレクトリ「work」中の「Graphene_STM.dat」。ソフトウェアWSxM [153]を使い、cubeファイル「Graphene_STM.pden.cube」を等値面値0.0001で可視化。

Slackでの議論

WSxMのtwitterがあります。

https://twitter.com/wsxm_official

ここからlinkが張られています。

<http://www.wsxm.eu/download.html>

ここでWSxM v5.0 Develop 10.2をdownloadして下さい。

こちらのversionしか、Gaussian cubeファイルをサポートしていません。

プルタブメニューにおいて File -> Open でFile Typeを Gaussian Cube Fileを指定して、該当するファイルを読み込んで下さい。ファイルが読み込まれたら青い画面をクリックして下さい(添付ファイルを参考にして下さい)。

次に Process -> Filter -> Create STM image として、さらに Apply を押せばSTM像が出てきます。

VESTAでも切り出した面上の等高線図が描画できるかと思いませんので、同じような図が書けると思っています。

マニュアル中でGraphene_STM.dat と記載されておりますが、work中に存在しているファイル名は**Graphite_STM.dat**です。

内殻束縛エネルギーを計算する

アセチレン分子のC-1sの絶対束縛エネルギーの計算

アセチレン分子(C₂H₂)の内殻準位(C-1s)の絶対束縛エネルギーを計算し、実験値と比較する。

http://www.openmx-square.org/openmx_man3.9jp/node192.html

1. 始状態の計算 \$mpirun -np 4 ./openmx C2H2.dat | tee C2H2.std
 C2H2.dat中で level.of.fileout 1 と指定しておく。

2. 終状態の計算 \$mpirun -np 4 ./openmx C2H2-CH.dat | tee C2H2-CH.std
 C2H2-CH.dat中で level.of.fileout 1 と指定しておく。

3. エネルギー差の計算

outファイルから全エネルギー値を取得し、エネルギー差を計算する。

$$\begin{aligned} E_b^{(\text{gas})} &= E_f^{(0)}(N-1) - E_i^{(0)}(N), \\ &= -66.084858926233 - (-76.787732114928) = 10.702873188695(\text{Hartree}), \\ &\approx 291.24(\text{eV}). \end{aligned}$$

実験値は 291.14 eV

始状態と終状態 (アセチレン分子)の差電子密度解析

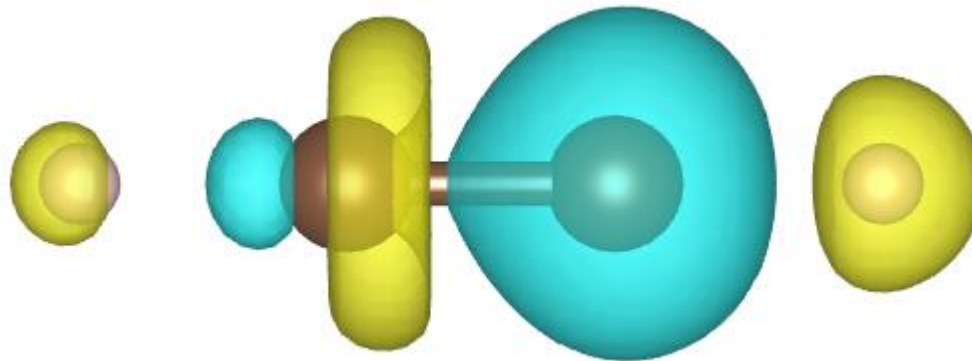
アセチレン分子(C₂H₂)の始状態と終状態の差電子密度を以下のwebに記載された手順に従い、解析する。

http://www.openmx-square.org/openmx_man3.9jp/node209.html

- (1) source中でdiff_gcubeをコンパイルし、workにコピーする。
- (2) diff_gcubeを用いて、始状態と終状態の差電子密度を計算する

```
$ ./diff_gcube C2H2.tden.cube C2H2-CH.tden.cube C2H2-diff.tden.cube
```

- (3) VESTAを用いて、C2H2-diff.tden.cubeを可視化する。



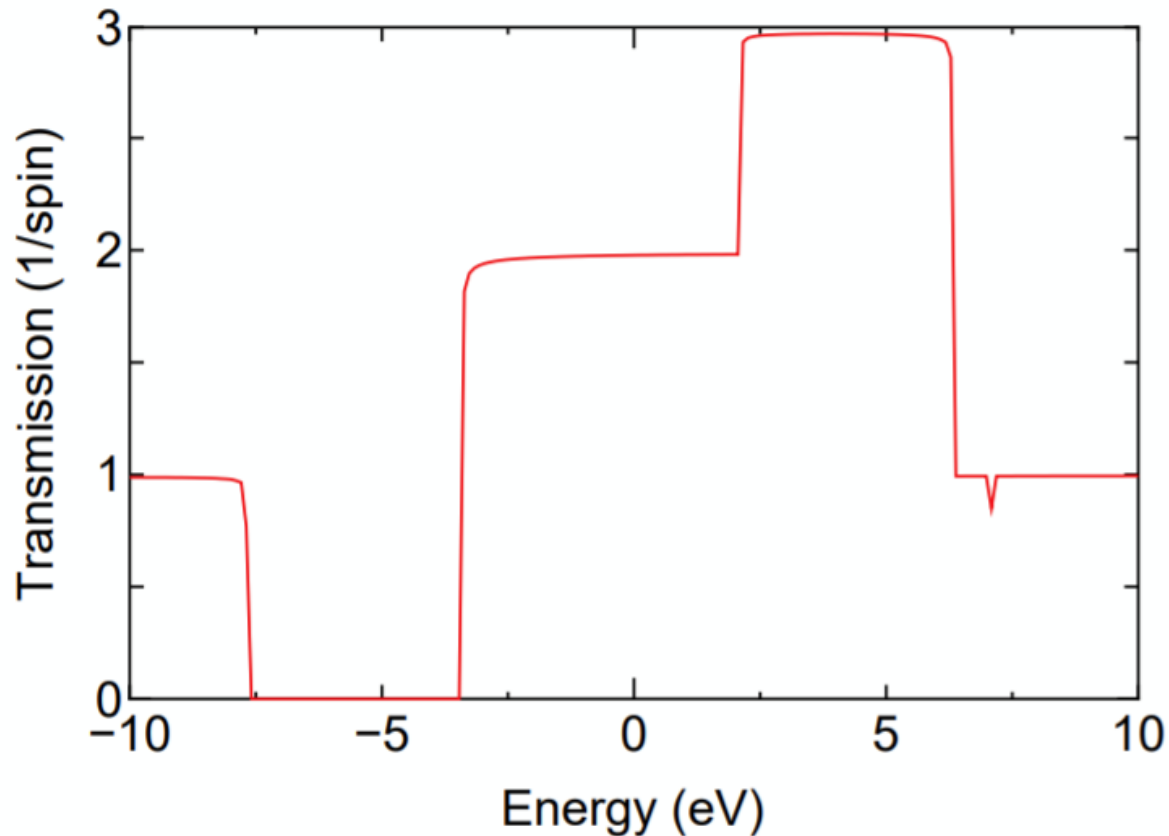
非平衡グリーン関数法を用いて透過率を計算する

非平衡グリーン関数法による透過率計算

以下のwebの手順に従い、非平衡グリーン関数法を用いて炭素鎖の透過率を計算する。

http://www.openmx-square.org/openmx_man3.9jp/node127.html

計算が正常に実行されれば、透過率のエネルギー依存性は以下のようなになる。



計算で留意すべき点

- runttest(L)によるベンチマーク計算
- カットオフエネルギーの選択
- エネルギーカーブの計算
- SCF計算の理論と実際
- 構造最適化の理論と実際
- 基底関数の選択
- 表面状態やfloating状態
- リスタート
- バイナリモードでの出力

runtest(L)によるベンチマーク計算

OpenMXのインストールを正常に終了したとしても、プログラムが正常動作しない場合がある。この問題はコンパイラオプション(-O3)や数値演算ライブラリから、生じていることが多い。

runtestを実行することで、一通りの機能が正常に動作していることが確認できる。

For the MPI parallel running

```
% mpirun -np 8 openmx -runtest
```

For the OpenMP/MPI parallel running

```
% mpirun -np 8 openmx -runtest -nt 2
```

結果はruntest.resultに出力され、保存されている結果(全エネルギー、力)の差分が分かる。

1	input_example/Benzene.dat	Elapsed time(s)= 4.78	diff Utot= 0.000000000000	diff Force= 0.000000000002
2	input_example/C60.dat	Elapsed time(s)= 14.96	diff Utot= 0.000000000019	diff Force= 0.000000000004
3	input_example/CO.dat	Elapsed time(s)= 9.86	diff Utot= 0.000000000416	diff Force= 0.000000000490
4	input_example/Cr2.dat	Elapsed time(s)= 10.70	diff Utot= 0.000000000000	diff Force= 0.000000000044
5	input_example/Crys-MnO.dat	Elapsed time(s)= 19.98	diff Utot= 0.000000004126	diff Force= 0.000000001888
6	input_example/GaAs.dat	Elapsed time(s)= 26.39	diff Utot= 0.000000001030	diff Force= 0.000000000007
7	input_example/Glycine.dat	Elapsed time(s)= 5.48	diff Utot= 0.000000000001	diff Force= 0.000000000000
8	input_example/Graphite4.dat	Elapsed time(s)= 5.00	diff Utot= 0.000000002617	diff Force= 0.0000000015163
9	input_example/H2O-EF.dat	Elapsed time(s)= 4.88	diff Utot= 0.000000000000	diff Force= 0.000000000113
10	input_example/H2O.dat	Elapsed time(s)= 4.60	diff Utot= 0.000000000008	diff Force= 0.0000000013375
11	input_example/HMn.dat	Elapsed time(s)= 13.44	diff Utot= 0.000000000001	diff Force= 0.000000000001
12	input_example/Methane.dat	Elapsed time(s)= 3.64	diff Utot= 0.000000000001	diff Force= 0.000000002263
13	input_example/Mol_MnO.dat	Elapsed time(s)= 9.43	diff Utot= 0.000000003714	diff Force= 0.000000000540
14	input_example/Ndia2.dat	Elapsed time(s)= 5.67	diff Utot= 0.000000000004	diff Force= 0.000000000001

Total elapsed time (s) 138.79

runtestLは大きなジョブによるテスト。マシン性能を測る上で便利。様々な計算機上での結果が work/input_example 及び work/large_example 中にある。

カットオフエネルギーの選択

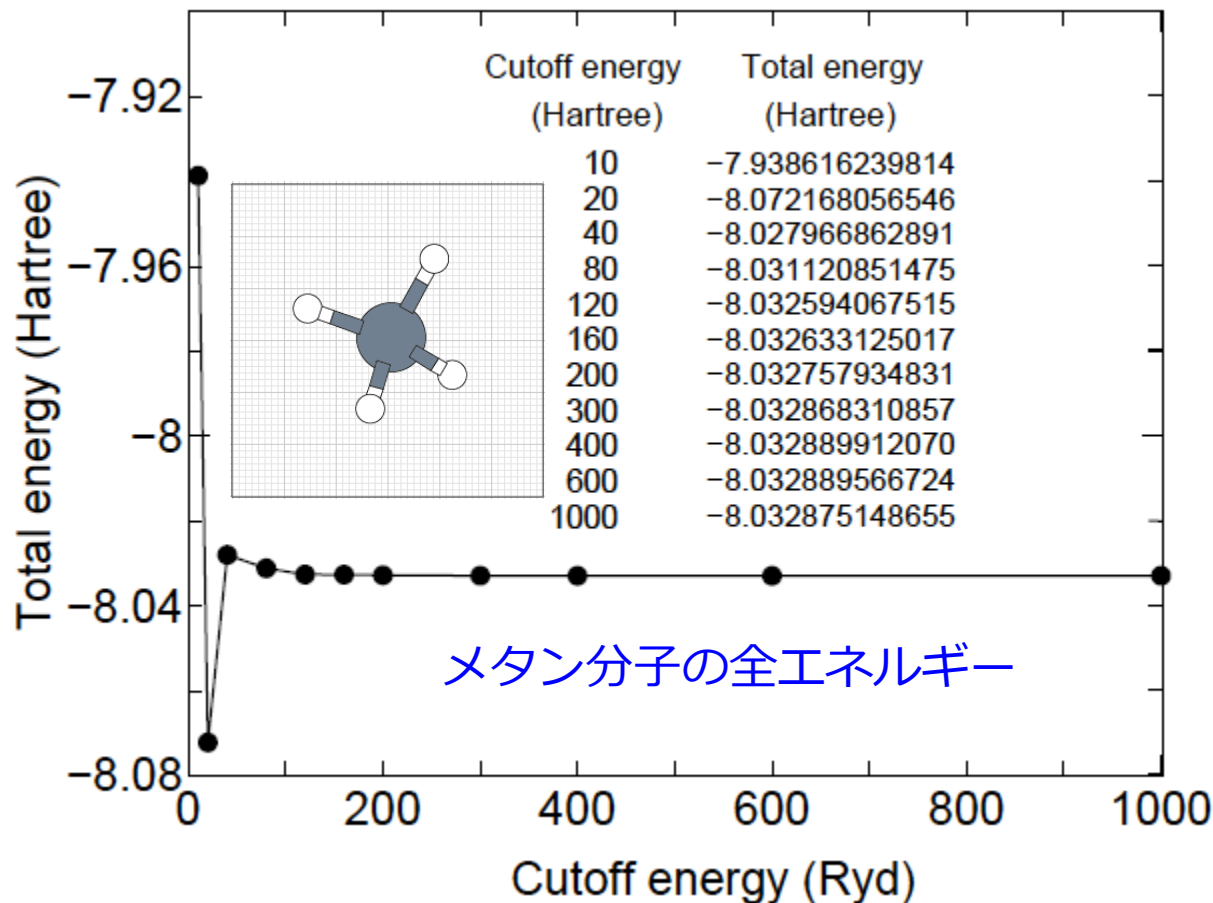
scf.energycutoff 200 # default=150 Ryd

実空間を離散化するFFTグリッドはscf.energycutoffで指定する。FFTグリッドはPoisson方程式の解法、 $E_{\delta_{ee}}$, E_{xc} の計算に使用する。

多くの系で200 Ryd程度で収束する。

擬ポテンシャルにセミコア状態を含む場合には、大きめのカットオフを取った方が安全。300-400 Ryd. 程度。

メモリの増加量:
 $O(E^{3/2})$



カットオフエネルギーの選択

H₂Oの最適化構造

水分子の最適化構造のscf.energycutoffに対する依存性。
180Ryd.でほぼ収束していることが分かる

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

エネルギーカーブの計算

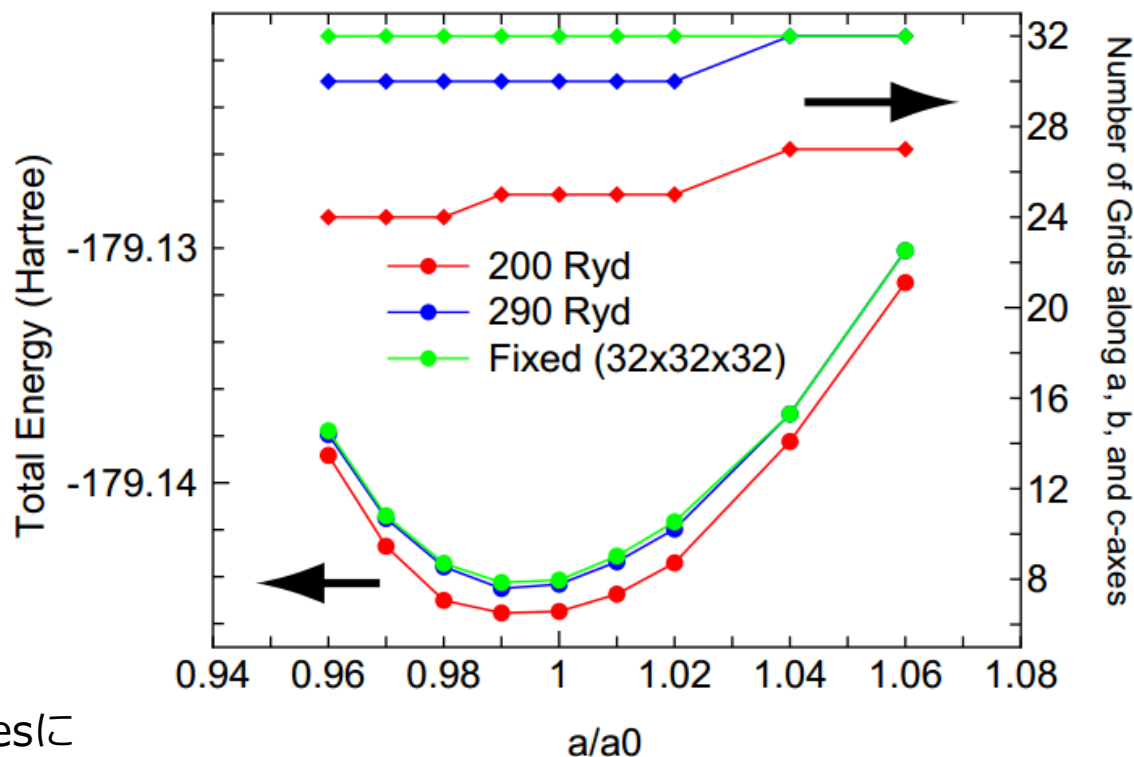
格子定数を変化させてエネルギー曲線を計算するには以下のkeywordが使用できる。

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

scf.energycutoffによってグリッドを決定すると、格子定数の増加に応じて、不連続にグリッド数が変化する。このグリッド数の突然の変化はエネルギーカーブに不連続なジャンプを生じさせる。これを避けるためにはグリッド数を固定して計算を行うべきである。次のkeywordで明示的にグリッド数が指定できる。

scf.Ngrid 32 32 32

それぞれの数字はa-, b-, and c-axesに対するグリッド数。scf.energycutoffとscf.Ngridをどちらも指定した場合にはscf.Ngridが優先される。



MD.Evslc.flag 1 1 0 # a, b, c-軸に対して
1: 格子定数を変化, 0: 格子定数を変化させない。

SCF計算：電子密度の混合

KS有効ポテンシャルは ρ の関数である。
しかし本来、 ρ はKS方程式から求められる。

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

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

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

単純混合法

入力と出力の電子密度を単純に足し合わせて、次の入力電子密度とする。

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

大きなバンドギャップのある系、小さい系に有効な方法

OpenMXで利用可能なSCF収束法

Simple mixing (Simple)

関連するkeywords: scf.Init.Mixing.Weight,
scf.Min.Mixing.Weight, scf.Max.Mixing.Weight

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

関連するkeywords: scf.Init.Mixing.Weight,
scf.Min.Mixing.Weight, scf.Max.Mixing.Weight,
scf.Mixing.History, scf.Mixing.StartPulay

Guaranteed reduction Pulay method (GR-Pulay)

関連するkeywords: scf.Init.Mixing.Weight,
scf.Min.Mixing.Weight, scf.Max.Mixing.Weight,
scf.Mixing.History, scf.Mixing.StartPulay

Kerker mixing (Kerker)

関連するkeywords: scf.Init.Mixing.Weight,
scf.Min.Mixing.Weight, scf.Max.Mixing.Weight,
scf.Kerker.factor

RMM-DIIS with Kerker metric (RMM-DIISK)

関連するkeywords: scf.Init.Mixing.Weight,
scf.Min.Mixing.Weight, scf.Max.Mixing.Weight,
scf.Mixing.History, scf.Mixing.StartPulay,
scf.Mixing.EveryPulay, scf.Kerker.factor

- 5つのSCF収束法が利用可能。
- 多くの系でRMM-DIISKがベスト。
- RMM-DIIS、RMM-DIISKはどちらもPulay混合法であるが、前者は密度行列、後者は電子密度のフーリエ変換に対してPulay混合を適用する。
- 非常に収束が難しい系はKerkerがベター。
- plus Uの場合にはRMM-DIISの方がRMM-DIISKより収束性に優れることもしばしばある。

RMM-DIISKによるSCF収束

アイデア

過去の(フーリエ空間での)残差ベクトルの線形和のノルムを最小化する。

$$\bar{R}_{n+1} = \sum_{m=n-(p-1)}^n \alpha_m R_m, \quad R_n(\mathbf{q}) \equiv \tilde{n}_n^{(\text{out})}(\mathbf{q}) - \tilde{n}_n^{(\text{in})}(\mathbf{q}),$$

$$F = \langle \bar{R}_{n+1} | \bar{R}_{n+1} \rangle - \lambda \left(1 - \sum_m^n a_m \right), \quad \langle R_m | R_{m'} \rangle \equiv \sum_{\mathbf{q}} \frac{R_m^*(\mathbf{q}) R_{m'}(\mathbf{q})}{w(\mathbf{q})},$$

$$= \sum_{m,m'} \alpha_m \alpha_{m'} \langle R_m | R_{m'} \rangle - \lambda \left(1 - \sum_m^n a_m \right). \quad \text{Kerker factor } w(\mathbf{q}) = \frac{|\mathbf{q}|^2}{|\mathbf{q}|^2 + q_0^2},$$

Fの最小化条件から次式が得られる。

$|\mathbf{q}|$ が小さい成分(長波長成分)に重きを置く。

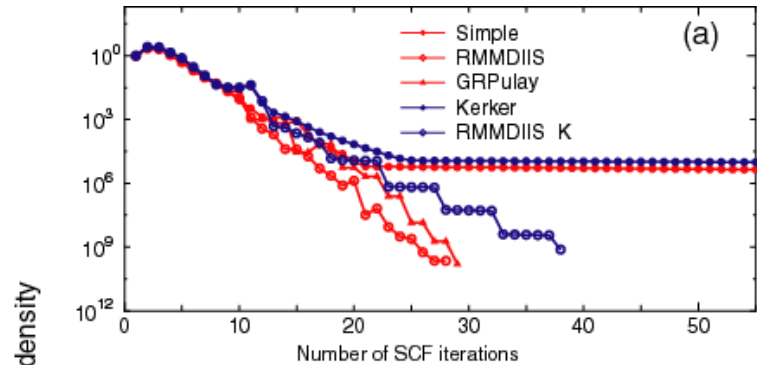
$$\frac{\partial F}{\partial \alpha_k} = 0 \quad \begin{pmatrix} \langle R_{n-(p-1)} | R_{n-(p-1)} \rangle & \cdots & \cdots & 1 \\ \cdots & \cdots & \cdots & 1 \\ \cdots & \cdots & \langle R_n | R_n \rangle & \cdots \\ 1 & 1 & \cdots & 0 \end{pmatrix} \begin{pmatrix} \alpha_{n-(p-1)} \\ \alpha_{n-(p-1)+1}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

“最適”な入力電子密度は次式で与えられる。

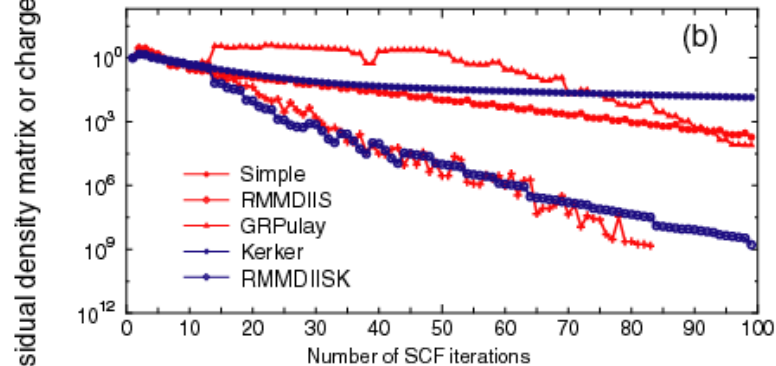
$$\rho_{n+1}^{(\text{in})} = \sum_{m=n-(p-1)}^n \alpha_m \rho_m^{(\text{in})} + \beta \sum_{m=n-(p-1)}^n \alpha_m R_m$$

様々な電荷密度混合法の比較

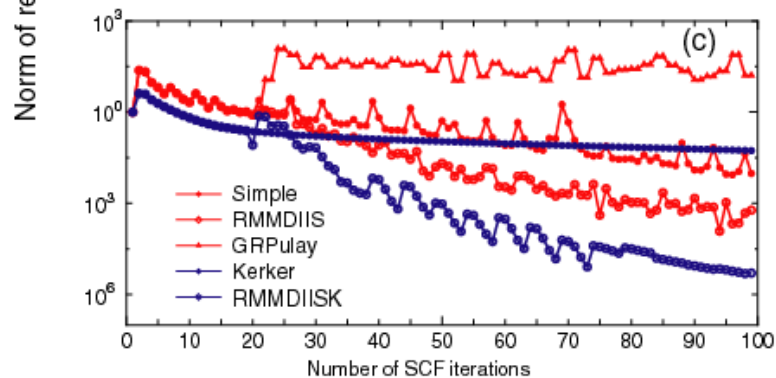
シアリック酸



Pt₁₃ クラスタ



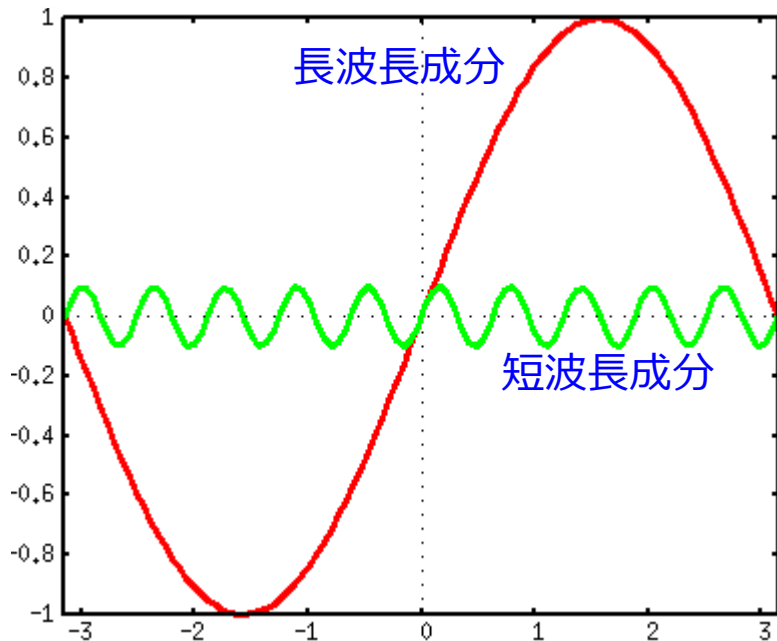
Pt₆₃ クラスタ



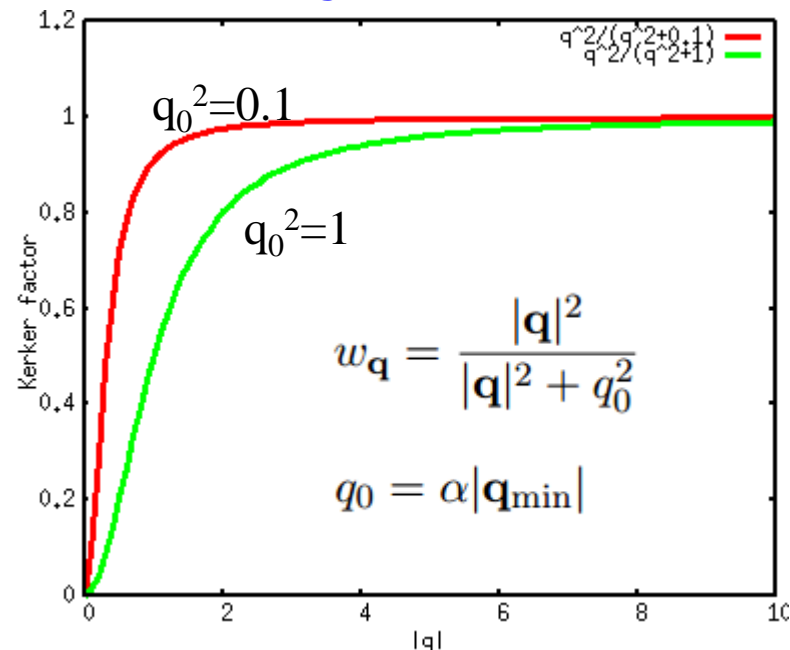
いずれの場合も
RMM-DIISもしくは
RMM-DIISKが
高速である。

Charge sloshingとKerker因子

SCF計算の際の大きな電子密度の揺らぎは密度の長波長成分(小さいq)から生じる



長波長成分から生じるcharge sloshingを低減するために、Kerker因子を導入する。 q_0 が大きいほど、sloshingが抑えられる。



Kerker因子 α はscf.Kerker.factor によって明示的に与えられるが、このkeywordを指定しない場合には自動で適切な値を推定する。

$$\alpha = \frac{0.5}{|\mathbf{b}_{\min}|^2} \left(4 \frac{Dq}{Aq} + 1.0 \right)$$

$$Aq = \frac{1}{3} \left(|\mathbf{b}_1|^2 + |\mathbf{b}_2|^2 + |\mathbf{b}_3|^2 \right),$$

$$Dq = \frac{1}{3} \sum_{i < j} \left| |\mathbf{b}_i|^2 - |\mathbf{b}_j|^2 \right|,$$

推定値は標準出力に以下の様に表示されるので、この値を手掛かりに調整すると良い。

Automatic determination of Kerker_factor: ****

RMM-DIISKのkeywords

RMM-DIISKは以下のkeywordsで制御される。

scf.Init.Mixing.Weight	0.01
scf.Min.Mixing.Weight	0.01
scf.Max.Mixing.Weight	0.2
scf.Mixing.History	40
scf.Mixing.StartPulay	20
scf.Mixing.EveryPulay	1
scf.Kerker.factor	5.0

- scf.Mixing.StartPulayはRMM-DIISKをスタートするSCFステップを指定する。
- それ以前のステップはKerker法が適用される。RMM-DIISK法は準ニュートン法であるから、ある程度の収束が達成された後に、乗り換えるべきである。さもないと収束しない場合がある。
- RMM-DIISKにおける履歴数はscf.Mixing.Historyで指定する。
- scf.Init.Mixing.Weight, scf.Min.Mixing.Weight, scf.Max.Mixing.Weight はKerker法を適用される際の混合比の初期値、最小値、最大値である。収束状況をモニターし、この範囲で自動的に調整される。
- Kerker因子はscf.Kerker.factorで与えられる。指定しない場合には自動的に最適値が推定され、使用される。
- RMM-DIISK法に乗り換えた後に、RMM-DIISKによる混合を実施する頻度をscf.Mixing.EveryPulayで指定する。1の場合には毎回、実施することに対応。

LDA+UにおけるSCF収束

LDA+U法のエネルギー表式

$$E = E_{\text{DFT}}[\rho] + E_U[\mathbf{n}]$$

$$E_U = \frac{1}{2} \sum_{\alpha} (U_{\alpha} - J_{\alpha}) \sum_{\sigma} \{ \text{Tr}(n_{\alpha}^{\sigma}) - \text{Tr}(n_{\alpha}^{\sigma} n_{\alpha}^{\sigma}) \}.$$

- LDA+U法のエネルギーは電子密度と密度行列の汎関数である。
- 電子密度 ρ に対する混合法, Kerker, RMM-DIISK法をLDA+U法に適用すると、密度行列 \mathbf{n} に対しては混合が行われずに、1ステップ前の密度行列 \mathbf{n} が使用されることになる。
- これは単純混合の100%混合に対応し、密度行列のsloshingが引き起こされる可能性がある。
- **そこでLDA+U法ではRMM-DIIS法の使用が望ましい。**
- 本来は電子密度と密度行列の混合を同時に実施する混合法を開発するべきであるが、現時点でのOpenMXには実装されていない。

他の電子密度混合法との比較

Anderson混合法

$$\bar{\rho}_n^{(\text{in})} = \rho_n^{(\text{in})} + \sum_{m=n-(p-1)}^{n-1} \theta_m (\rho_m^{(\text{in})} - \rho_n^{(\text{in})}), \quad \bar{R}_n^{(\text{in})} = R_n^{(\text{in})} + \sum_{m=n-(p-1)}^{n-1} \theta_m (R_m^{(\text{in})} - R_n^{(\text{in})}).$$

$$\rho_{n+1}^{(\text{in})} = \bar{\rho}_n^{(\text{in})} + \beta \bar{R}_n^{(\text{in})} \quad \rightarrow \text{equivalent to RMM-DIIS}$$

Broyden混合法

$$E = |G_n - G_{n-p}|^2 + \sum_{m=l+1-p}^{n-1} \langle \lambda_m | \{ (|n_{m+1}\rangle - |n_m\rangle) - G_n (|R_{m+1}\rangle - |R_m\rangle) \}$$

$$\frac{\partial E}{\partial G} = 0 \quad \frac{\partial E}{\partial \lambda} = 0$$

$$|n_{n+1}\rangle = |n_n\rangle - \sum_{m=n-p}^{n-1} \gamma_m^n (|n_{m+1}\rangle - |n_m\rangle) - G_{n-p} \left\{ |R_n\rangle - \sum_{m=n-p}^{n-1} \gamma_m^n (|n_{m+1}\rangle - |n_m\rangle) \right\}$$

$$G_{n-p} = -\beta \quad \rightarrow \quad \rho_{n+1}^{(\text{in})} = \bar{\rho}_n^{(\text{in})} + \beta \bar{R}_n^{(\text{in})} \quad \rightarrow \text{equivalent to RMM-DIIS}$$

RMM-DIIS, Anderson, Broyden 法は数学的には等価の方法であり、準Newton法に基づいている。

SCF収束性の改良に向けて

Broyden法

$$|n_{n+1}\rangle = |n_n\rangle - \sum_{m=n-p}^{n-1} \gamma_m^n (|n_{m+1}\rangle - |n_m\rangle) - G_{n-p} \left\{ |R_n\rangle - \sum_{m=n-p}^{n-1} \gamma_m^n (|n_{m+1}\rangle - |n_m\rangle) \right\}$$

$$G_{n-p} = -\beta \quad \rightarrow \quad \rho_{n+1}^{(\text{in})} = \bar{\rho}_n^{(\text{in})} + \beta \bar{R}_n^{(\text{in})}$$

もしGが保存できるならば、Broyden法がベストな方法であると考えられる。しかしながら、Gの行列サイズはq点に比例するため、保持しておくことは難しい。したがって現行のOpenMXにおいて、理論的な観点から**収束性を高める最も有効な方法は、履歴数を大きく取ること**である。

収束が難しい系に対しては30～50の履歴数をとると良い。

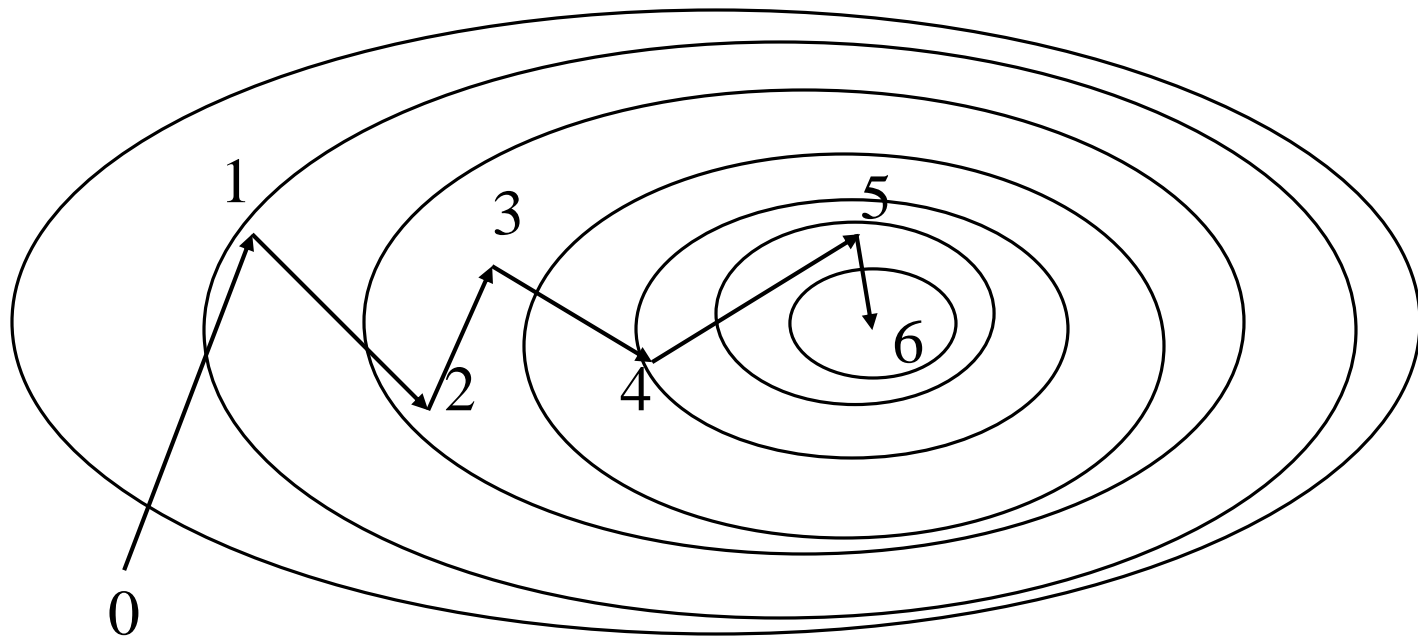
Broyden法の利点を生かしつつ、計算・メモリコストの小さい計算手法の開発が望まれる。L-BFGS法との組み合わせは有望であろう。

構造最適化

分子や固体の性質を精密に調べるためにはエネルギーの最小値を与える構造(最適構造)を計算する必要がある。

最急降下法(Steepest decent method)

エネルギーの勾配方向に少しずつ移動する。



あまり効率的ではない。

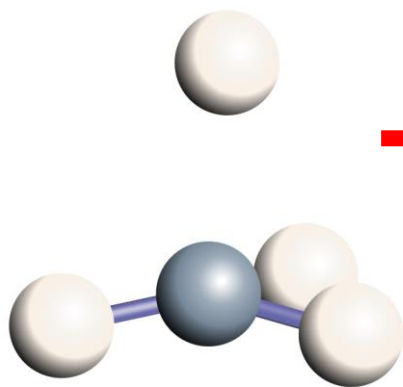
最急勾配法による構造最適化

メタン分子の炭素原子のx座標が
以下のように0.3 Åに変更

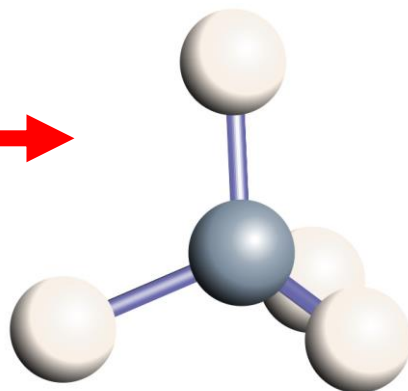
```
<Atoms.SpeciesAndCoordinates
 1 C  0.300000  0.000000  0.000000  2.0  2.0
 2 H -0.889981 -0.629312  0.000000  0.5  0.5
 3 H  0.000000  0.629312 -0.889981  0.5  0.5
 4 H  0.000000  0.629312  0.889981  0.5  0.5
 5 H  0.889981 -0.629312  0.000000  0.5  0.5
Atoms.SpeciesAndCoordinates>
```

Work中の「Methane2.dat」を
用いて本計算は再現できる。

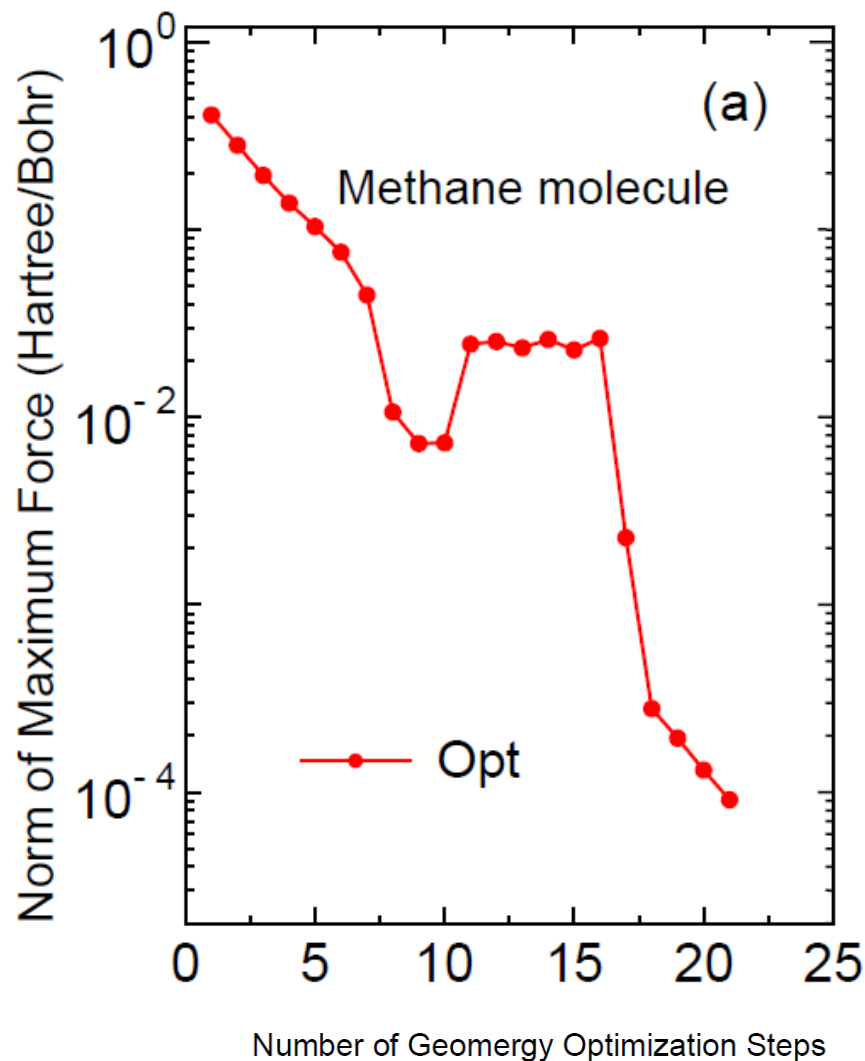
初期構造



最終構造



構造最適化の過程



解析に関する出力ファイル

met2.out中に構造最適化の履歴が確認できる。

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

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

met2.mdをOpenMX Viewerにドラッグ&ドロップすると構造最適化過程が容易に確認できる。

The screenshot shows a web browser window at www.openmx-square.org/viewer/index.html. The main content is a 3D ball-and-stick model of a molecule, likely a diatomic molecule with a central blue atom and two white atoms. A dialog box titled "Structural Change" is open, displaying "time= 0.000 (fs) Energy= -8.09572 (Hartree)" and "Frame 1/28". The dialog box contains navigation buttons: "First", "<<", "Stop", ">>", and "Last". Below these are fields for "Make animation" (Every 1 frames), "Delayed time (ms)" (0), and "File format" (not supp.). At the bottom of the browser window, there is a status bar with various settings: "Supercell" (1x1x1), "Atoms" (rendering), "Bonds" (fine), "Number" (off), "Symbol" (off), "BG" (black), "Atom Size" (1.0), "Bond Thickness" (1.0), "Bond Factor" (1.0), "Cells" (off), "Axes" (off), "Perspective" (off), "Structure" (off), "Dynamics" (on), "Net Charge" (off), "Force" (off), "Velocity" (off), "Rot on x" (0), "Rot on y" (0), "Rot on z" (0). The status bar also includes the text "OpenMX website" and "Rotate: click+drag Zoom: scroll Translate: ctrl+click+drag Note: click the middle button for Safari".

ニュートン法による構造最適化

エネルギーをTaylor展開する

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

座標の微分を計算

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

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

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

これを $\Delta \mathbf{x}$ について解けば、

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

もしHessian行列Hが求められるならば、この方法は効率的である。
しかし一般の場合に、Hを求めることは困難である。

準ニュートン法による構造最適化

OpenMXにおける構造最適化は準ニュートン法に基づいている。
OpenMX Ver. 3.7においては4つの最適化方法が実装されている。

$$\mathbf{r}_{\text{new}} = \mathbf{r}_{\text{DIIS}} + \Delta\mathbf{r}$$

$$\Delta\mathbf{r} = -H^{-1}\mathbf{g}_{\text{DIIS}}$$

近似ヘシアン行列Hの計算手法

DIIS

BFGS

RF(rational function)

EF(eigenvector following)

H=I

BFGS

BFGS+RF

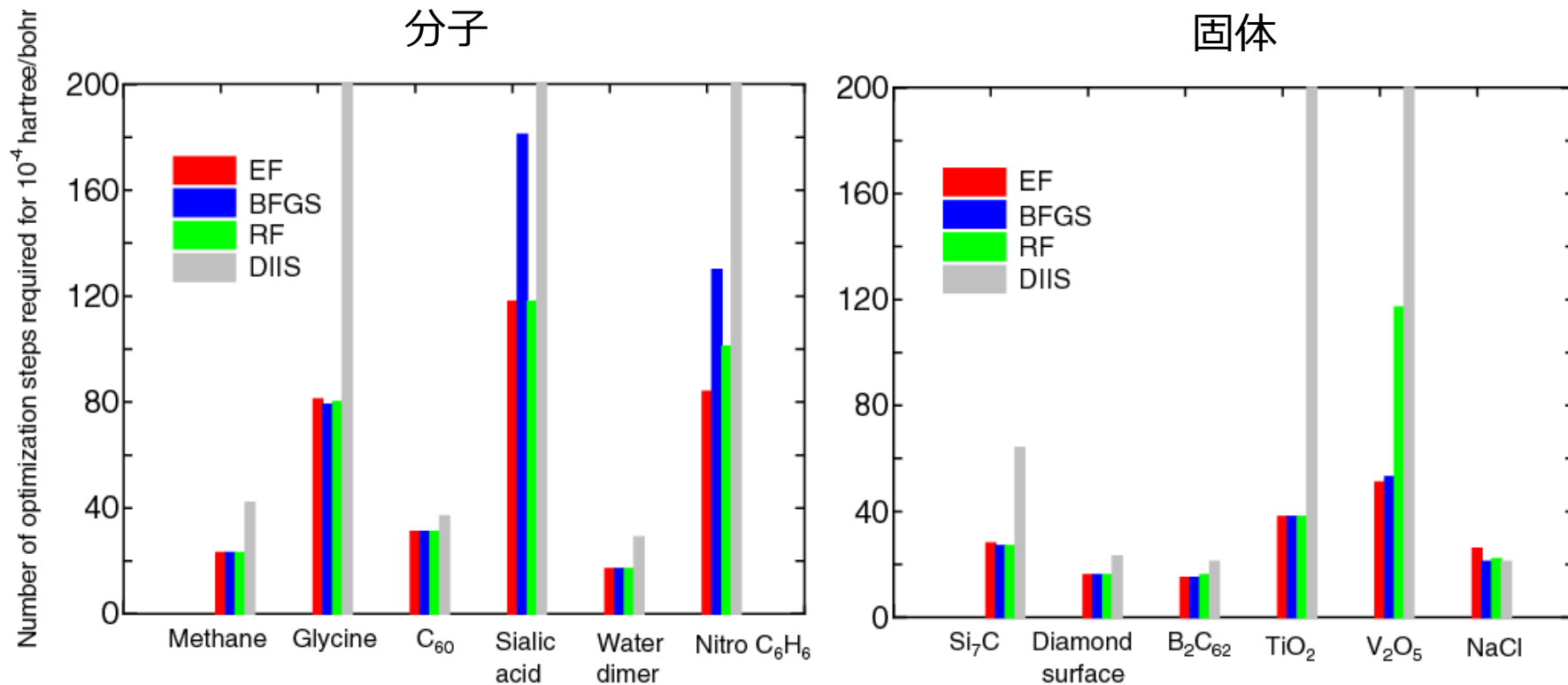
BFGS plus monitoring
of eigenvalues of H

Broyden-Fletcher-Goldfarb-Shanno (BFGS) 法

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

赤枠で囲まれた内積が正であるならば、Hの正定値性は保たれる。

構造最適化法の比較



- ベンチマーク計算からはEF法とRF法の性能が良いことが分かる。
- 構造に自由度が大きい分子系は収束しにくい。

このベンチマーク計算の入カファイルと出カファイルは以下のディレクトリにある。
openmx3.7/work/geoopt_example".

構造最適化に関連するkeywords

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)

特に、これらの準ニュートン法に基づいた構造最適化法は次の2種のキーワードにより制御することができます。

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

キーワード「MD.Opt.DIIS.History」は近似ヘシアンを更新するための過去の履歴ステップ数を指定します。デフォルト値は3です。また、「EF」、「BFGS」、「RF」、「DIIS」を開始する構造最適化ステップをキーワード「MD.Opt.StartDIIS」により指定します。これらの方法を開始する前の構造最適化ステップは、最急降下法により実行されます。デフォルト値は5です。

エネルギーカーブの計算

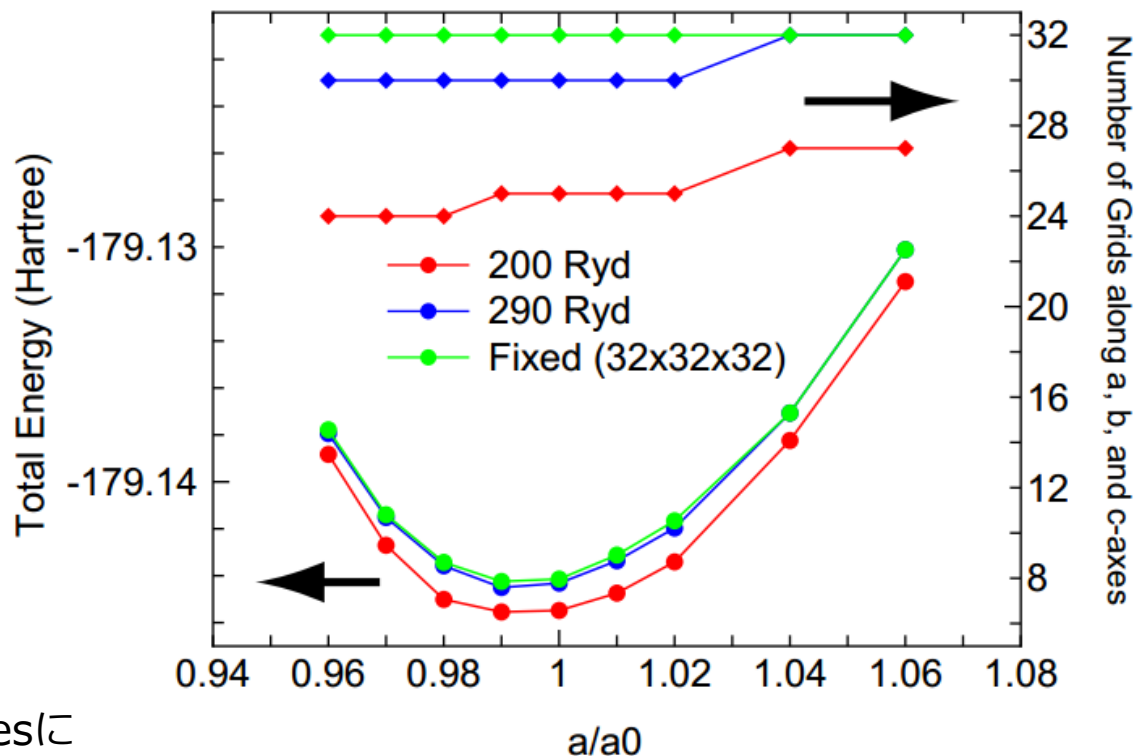
格子定数を変化させてエネルギー曲線を計算するには以下のkeywordが使用できる。

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

scf.energycutoffによってグリッドを決定すると、格子定数の増加に応じて、不連続にグリッド数が増える。このグリッド数の突然の変化はエネルギーカーブに不連続なジャンプを生じさせる。これを避けるためにはグリッド数を固定して計算を行うべきである。次のkeywordで明示的にグリッド数が指定できる。

scf.Ngrid 32 32 32

それぞれの数字はa-, b-, and c-axesに対するグリッド数。scf.energycutoffとscf.Ngridをどちらも指定した場合にはscf.Ngridが優先される。



MD.EvsLC.flag 1 1 0 # a, b, c-軸に対して
1: 格子定数を変化, 0: 格子定数を変化させない。

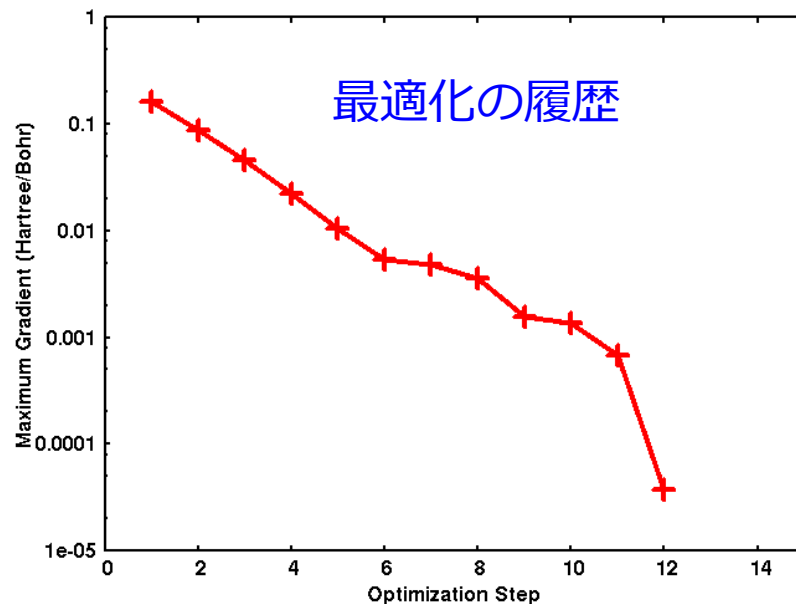
格子定数・内部座標の最適化

ダイヤモンド格子の内部座標を少しずらした所からスタートする。

```
Atoms.Number          2
Atoms.SpeciesAndCoordinates.Unit  frac # Ang|AU
<Atoms.SpeciesAndCoordinates
  1  C  0.10000000000000  0.00000000000000  -0.050000000000000  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>
```

最適化に関するkeywords:

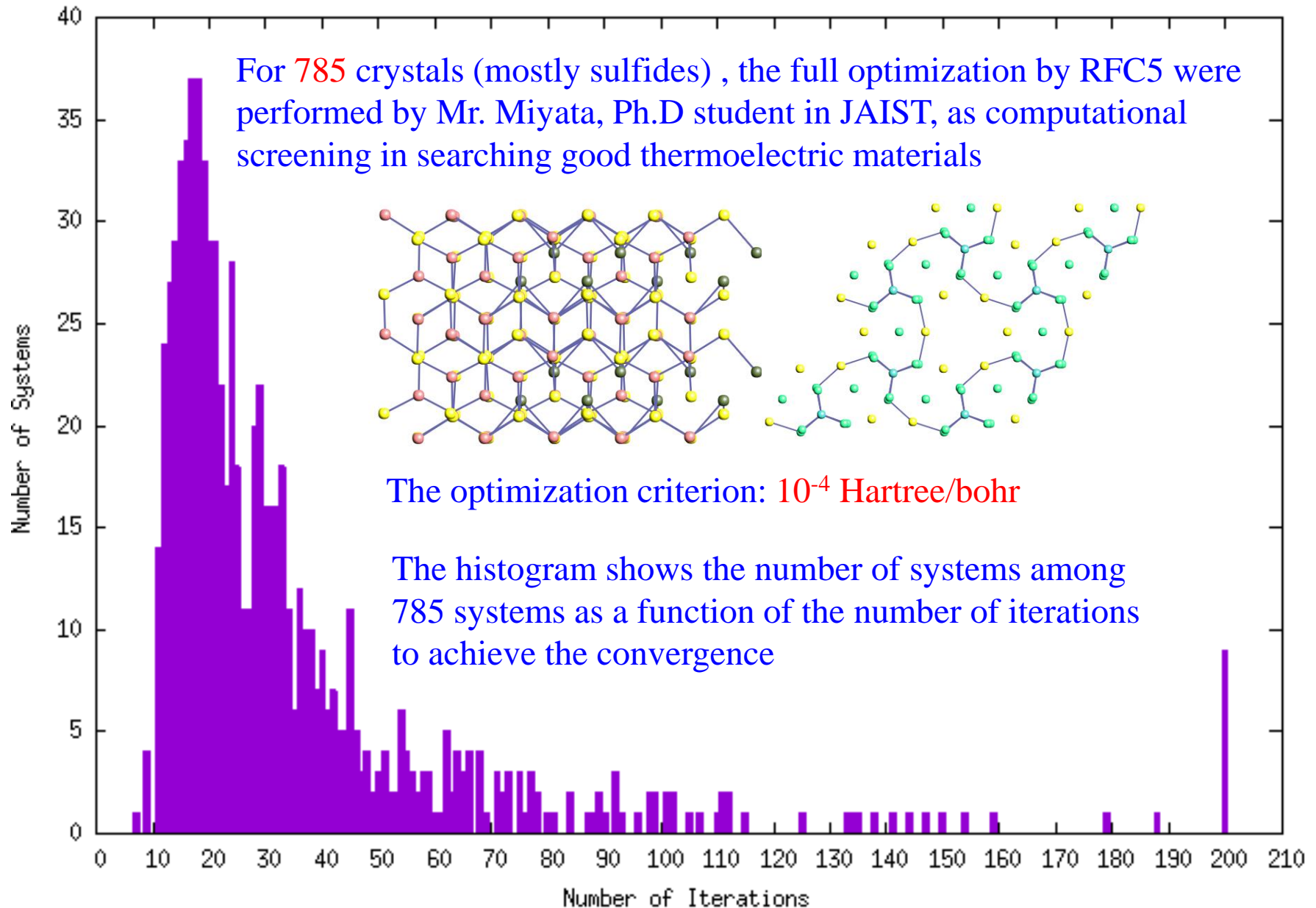
```
MD.Type          RFC5
MD.Opt.DIIS.History  3
MD.Opt.StartDIIS   7
MD.Opt.EveryDIIS   100000
MD.maxIter        100
MD.TimeStep       1.0
MD.Opt.criterion   0.0003
```



この計算はwork中のCdia-RF5.datで追試できる。

http://www.openmx-square.org/openmx_man3.8/node56.html も参照のこと

RFC5のベンチマーク計算



基底関数の指定

基底関数は以下の様に指定する。

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

- O7.0はO7.0.paoに対応する。
- -s2p2d1は動径関数として、s軌道を二つ、p軌道を2つ、d軌道を1つ選択することを意味する。
- この時、酸素Oに対して $2 \times 1 + 2 \times 3 + 1 \times 5 = 13$ 基底が割り当てられることになる。

O7.0.paoやO_PBE19.vpsの保存先は以下のkeywordで指定する。

```
DATA.PATH ../DFT_DATA19 # default=../DFT_DATA19
```

Default値は../DFT_DATA19である。

基底関数の選択: H₂Oの場合

Database(2013)においてH7.0.paoとO7.0.paoをクリックすると以下の情報が得られる。
http://www.jaist.ac.jp/~t-ozaki/vps_pao2013/H/index.html http://www.jaist.ac.jp/~t-ozaki/vps_pao2013/O/index.html

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

H7.0.pao

```
Eigenvalues  
Lmax= 3 Mul=15  
| mu 0 0 -0.23595211038442  
| mu 0 1 0.14109389991827  
| 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  
| 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
```

↖3

↖6

↖5

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

O7.0.pao

```
Eigenvalues  
Lmax= 3 Mul=15  
| mu 0 0 -0.87913976280231  
| mu 0 1 0.06809061901229  
| 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  
| mu 1 0 -0.33075182895384  
| mu 1 1 0.16378499567753  
| 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  
| 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
```

↖1

↖4

↖2

↖7

↖8

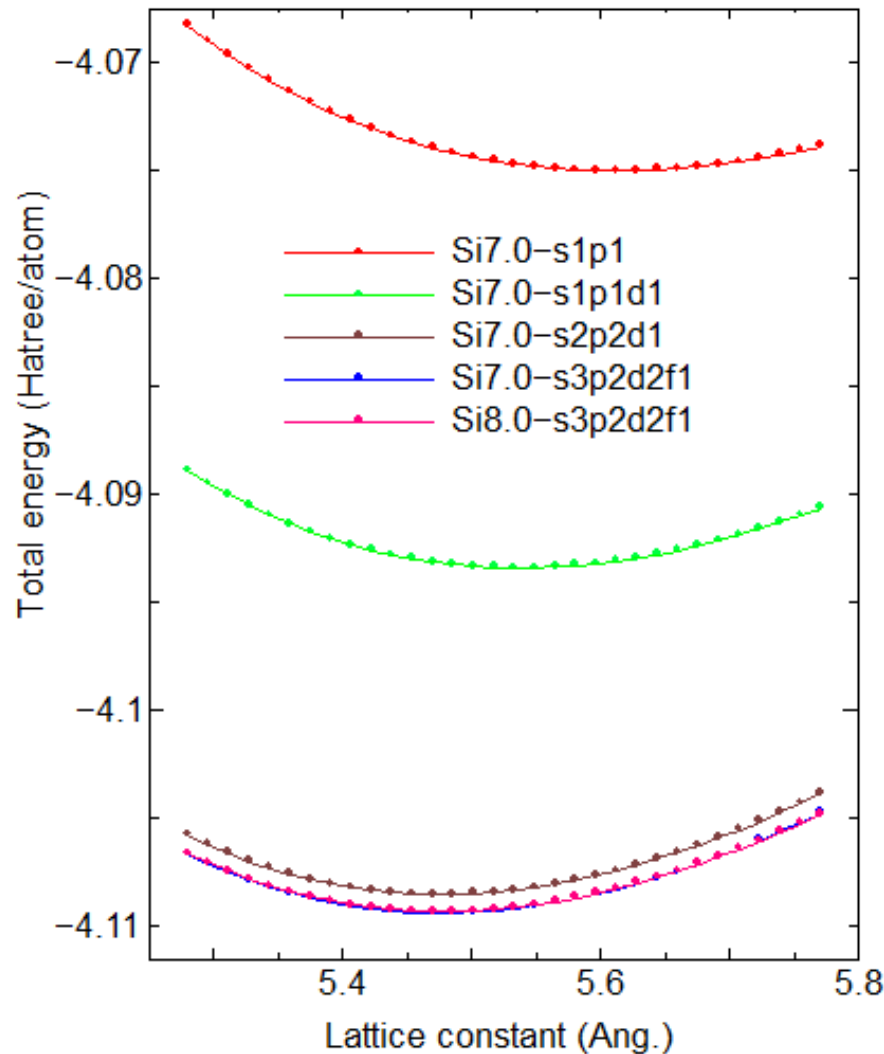
擬原子状態で低エネルギーの軌道から選択するとH7.0-s2p1, O7.0-s2p2d1となる。

基底関数の選択: Siの場合 (1)

Si7.0.pao

Eigenvalues		
Lmax= 3	Mul=15	
mu	0 0	-0.59320968623145
mu	0 1	0.01654247265872
mu	0 2	0.57915688461708
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	14.98369777897096
mu	0 12	17.62014479571136
mu	0 13	20.47011746372617
mu	0 14	23.53245309073866
mu	1 0	-0.31460013133101
mu	1 1	0.12937640798489
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
mu	2 1	0.35893514996065
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
mu	3 1	0.79082836114569
mu	3 2	1.52992065308726
mu	3 3	2.52537261496132

Si7.0.pao中の固有エネルギーの低い方から順次、基底に含める。

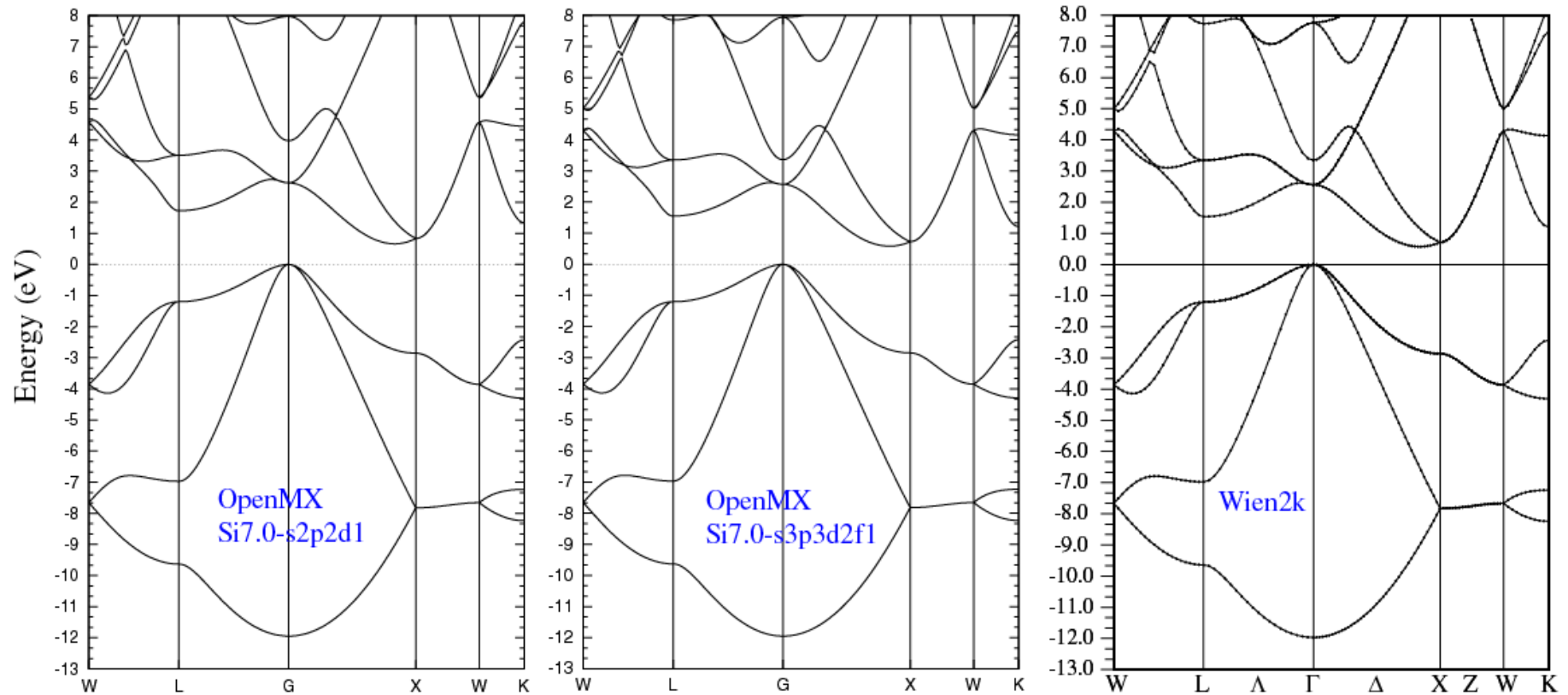


構造特性に関しては、Si7.0-s2p2d1で十分である。

Si7.0-s3p2d2f1とSi8.0-s3p2d2f1の比較から、カットオフ半径は7.0(a.u.)で収束していることが分かる。

基底関数の選択: Siの場合 (2)

バンド構造の比較からもSi7.0-s2p2d1でほぼ収束していることが分かる。



より完全な収束解を得るためにはSi7.0-s3p2d2f1(Si7.0-s3p3d2f1)を用いた方が良いが、大規模系を取り扱う場合や、広範囲な構造探索を行うためにはSi7.0-s2p2d1は精度とコストのバランスの取れた基底関数である。

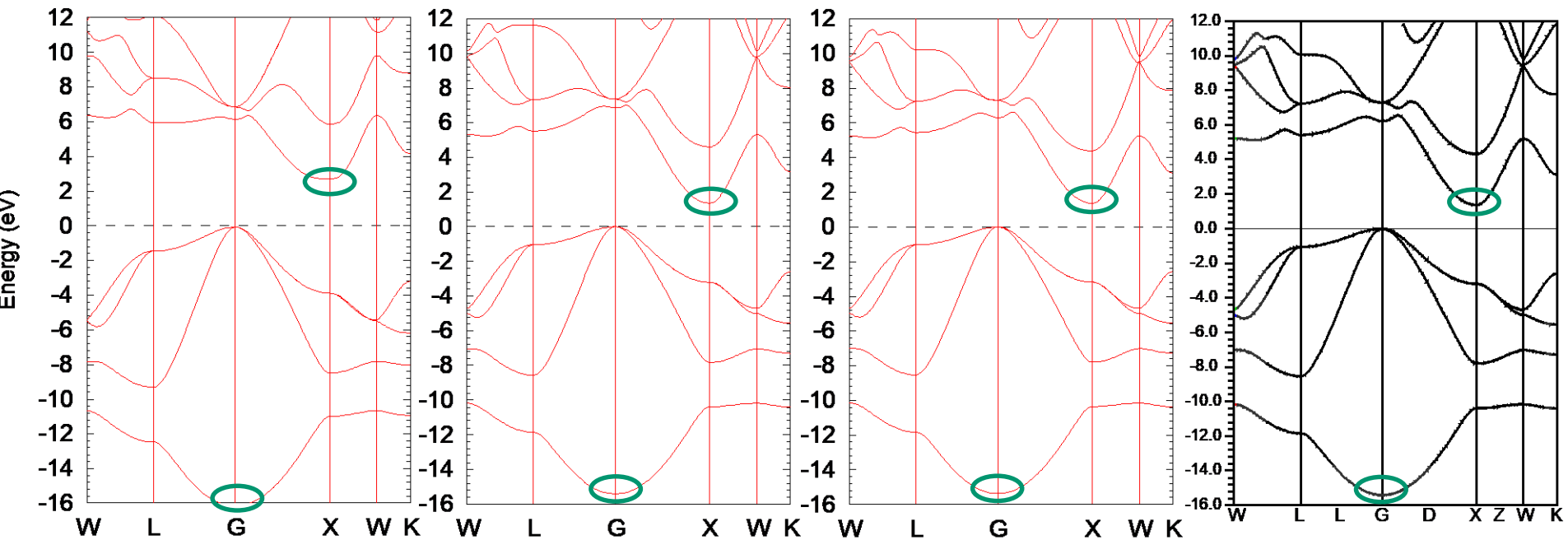
フローティング状態 in 3C-SiC

C6.0-s2p2
Si8.0-s2p2

C6.0-s2p2d1
Si8.0-s2p2d1

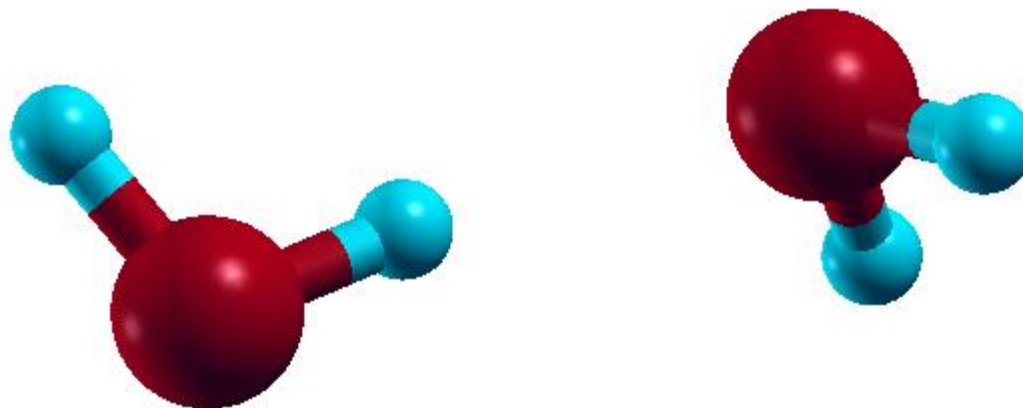
C6.0-s3p3d2f1
Si8.0-s3p3d2f1

Wien2k (FLAPW)



- ✓ d軌道を含めることが、バンド構造を再現する上で重要である。
- ✓ 少なくとも5eVまでのバンド構造はs2p2d1基底で再現されている。

基底関数重なり誤差 (BSSE)



	Equilibrium O-O distance (Ang.)	Dipole moment (Debye)	Binding energy (kcal/mol)	Binding energy (counterpoise 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	

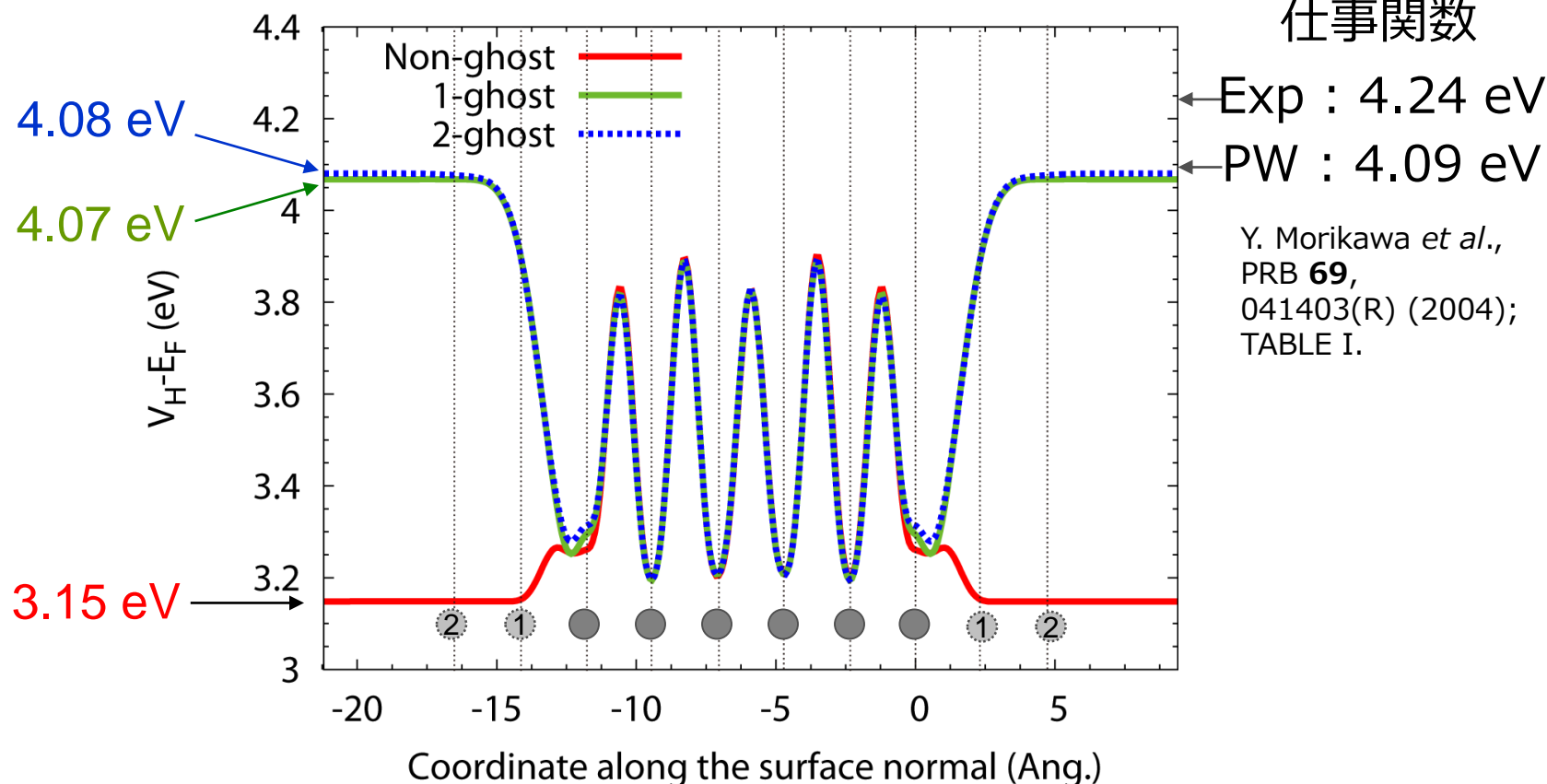
ベンチマーク計算から分子系に対するBSSEは ~ 0.5 kcal/mol

仕事関数の計算

fcc Al (111) surface

真空層にゴースト原子を置くことで仕事関数の高精度計算が可能である。
(本計算はDatabase(2006)の基底関数を用いて行われたため、
Database(2013)では状況が改善されている可能性がある。)

仕事関数



By Jippo and Ohfuti (Fujitsu)

Al(111) fcc 1x1 6L, GGA-PBE, Fix to the equilibrium lattice parameter
Al6.0-s1p2d1, Al_PBE (Database ver. 2006)
E6.0-s1p2d1, E (Database ver. 2006)

リスタート

- 計算が終了したにも関わらずSCFが収束していない場合、以下のkeywordを含めることで最終の電子密度分布を用いてSCF計算をリスタートすることが出来る。(ただしLDA+U法の場合には機能しない)

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

- 構造最適化や分子動力学計算が途中で終了した場合、最終座標や最終速度を含んだ入力ファイル(*.dat#)が利用できる。このリスタート用の入力ファイルを用いて、計算を途中からリスタート出来る。

バイナリモードでの出力

- 大規模な計算を実行し、複数のcubeファイルを出力する場合、ファイルシステムの性能によっては長時間を要する場合がある。この場合、以下のキーワードを用いてバイナリモードで出力することで出力時間を低減することが可能である。

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

- バイナリファイルはbin2txt.cを用いて、テキストファイルに変換できる。

```
コンパイル方法: gcc bin2txt.c -lm -o bin2txt
```

```
使用方法: ./bin2txt *.bin
```