



メモノート 2015
主にXANES計算を中心として

K.NAKADA



PWScf は Quantum ESPRESSO の計算部分のコード

(昔は PWScf だったが、もう少し大きな意味でのパッケージングで Quantum Espresso というパッケージなる)

GPL ライセンス

計算速度（だけ）比較、基本はほぼデフォルト値で
テスト (基底とかカットオフとかの比較を考えると**いい加減**)

16core	WIEN2k	VASP (high)	PWscf	OpenMX band	OpenMX cluster	GPAW PW	GPAW LCAO	CP2K
YTiO ₃ (k:444)	867.01	170.646		311.524		58.881	642.071	
Brookite (k:997)		478.646		1096.748	201.735	434.355		
C60 (k:111)	42590.01	74.388		19.267	18.64		20.367	
Graphene 1x1x1 (k:24241)	155.584	6.148		11.635	10.276	327.155	8.319	
H ₂ O mol	64223.02	8.581		6.086				
Methane		5.651		5.295	4.896	56.252 (1core)		
Nitro Benzene		59.518		14.911	7.585			

いろいろ、お手前のまとめ

構造最適化は使いにくい（対称性の都合）なのでお勧めしない

1) 拙作:構造変換系(VASP → PWscf)

PWvasp2in.py

2) 擬ポテンシャルの用意

PWSCF の基本インストール

```

DFLAGS      = -D__INTEL -D__FFTW3 -D__MPI -D__PARA -D__SCALAPACK $(MANUAL_DFLAGS)
FDFLAGS     = $(DFLAGS) $(MANUAL_DFLAGS)
IFLAGS      = -I../include
MOD_FLAG    = -I
MPIF90      = mpif90
CC          = icc
F77         = ifort
CPP         = cpp
CPPFLAGS    = -P -C -traditional $(DFLAGS) $(IFLAGS)
CFLAGS      = -O3 $(DFLAGS) $(IFLAGS) -I/opt/intel/mkl/include/fftw/
F90FLAGS    = $(FFLAGS) -nomodule -fpp $(FDFLAGS) $(IFLAGS) $(MODFLAGS)
FFLAGS      = -O2 -assume byterecl -g -I/opt/intel/mkl/include/fftw/
FFLAGS_NOOPT = -O0 -assume byterecl -g -traceback
FFLAGS_NOMAIN = -nofor_main
LD          = mpif90
LDFLAGS     = -static-intel
LD_LIBS     =
BLAS_LIBS   = -lmkl_intel_lp64 -lmkl_sequential -lmkl_core
BLAS_LIBS_SWITCH = external
LAPACK_LIBS =
LAPACK_LIBS_SWITCH = external
ELPA_LIBS_SWITCH = disabled
SCALAPACK_LIBS = -lmkl_scalapack_lp64 -lmkl_blacs_openmpi_lp64
FFT_LIBS    = -L/opt/intel/mkl/interfaces/fftw3xf -lfftw3xf_intel
MPI_LIBS    =
MASS_LIBS   =
AR          = ar
ARFLAGS     = ruv
RANLIB      = ranlib
FLIB_TARGETS = all
LIBOBJJS    = ../flib/ptools.a ../flib/flib.a ../clib/clib.a ../iotk/src/libiotk.a
LIBS        = $(SCALAPACK_LIBS) $(LAPACK_LIBS) $(FFT_LIBS) $(BLAS_LIBS) $(MPI_LIBS) $(MASS_LIBS) $(LD_LIBS)
WGET        = wget -O
PREFIX      = $(INSTALLDIR)

```

make.sys

FFTW と BLAS に注意

make.sys を用意出来たならば make する

configure は make.sys を作るので上書きされるので注意

make veryclean でも make.sys が消えるので注意

configure は 結構優秀で、MKL (BLAS,SCALAPACK) などは自動認識する(MKL の FFTW はビルドしていても自動認識はしないので注意)

make

to install, type at the shell prompt:

```
./configure  
make target
```

where target is one of the following:

pw	basic code for scf, structure optimization, MD
ph	phonon code, Gamma-only version and third-order derivatives
pwcond	ballistic conductance
neb	code for Nudged Elastic Band method
pp	postprocessing programs
cp	CP code: CP MD with ultrasoft pseudopotentials
ld1	utilities for pseudopotential generation
upf	utilities for pseudopotential conversion
tddfpt	time dependent dft code
gui	Graphical User Interface
pwall	same as "make pw ph pp pwcond neb"
all	same as "make pwall cp ld1 upf tddfpt"
xspectra	X-ray core-hole spectroscopy calculations
gipaw	NMR and EPR spectra
w90	Maximally localised Wannier Functions
want	Quantum Transport with Wannier functions
yambo	electronic excitations with plane waves
plumed	Metadynamics plugin for pw or cp
clean	remove executables and objects
veryclean	revert distribution to the original status
tar	create a tarball of the source tree
doc	build documentation
links	create links to all executables in bin/

SCF計算の入力 (準備)

注意)

入力ファイルにエラーがあると問答無用で止まる
どこでエラーが発生したか詳細には**絶対に教えてくれない**

 入力ミスの発見を延々と出来ない可能性がある

- (1)文字列はシングルクオートで囲む”
- (2)数値はそのまま
- (3)タグの区切りは基本, (カンマ) をつける。付けなくて動く場合もあるが、付けないと動かないものもある。
- (4)入力タグには**暗黙のうちに別のタグを要求しているものがある**ので、そのタグを用意しないと絶対に動かない
- (5)スペルミス等のPWscfが理解できないタグがあってもミス指摘してくれない。その時は原因不明で止まる。

構造作成

cif2cell がわりと万能 (python)

<http://cif2cell.sourceforge.net> (<http://sourceforge.net/projects/cif2cell/>)

Comp. Phy. Comm.182 (2011) 1183

cif2cell -p 変換プログラム -f 変換元INPUT

ヘルプ: cif2cell -h

- a) **VASP**: cif2cell -p **vasp** -f **case.cif** --vasp-cartesian-positions
- b) **ABINIT**: cif2cell -p **abinit** -f **case.cif** -o case.in
- c) **exciting**: cif2cell -p **exciting** -f **case.cif**
- d) **Fleur**: cif2cell -p **fleur** -f **case.cif**
- e) **elk**: cif2cell -p **elk** -f **case.cif**
- f) **CPMD**: cif2cell -p **cpmd** -f **case.cif** -o SrTiO3.inp
- g) **CASTEP**: cif2cell -p **castep** -f **case.cif** -o SrTiO3.cell
- h) **FHI-AIMS**: cif2cell -p **fhi-aims** -f **case.cif**
- i) **PWSCF**: cif2cell -p quantum-espresso -f case.cif

*) 個人には少々不満

PWvasp2in.py

POSCAR からの

構造変換+デフォルトコントロールファイル

ASE だと構造のみなのでめんどくさい
(デフォルトも含めて用意する)

+

using 自前のフレームワーク

```
&control
  calculation = 'scf',
  prefix = 'input',
  pseudo_dir = './',
  outdir = './work/',
/
&system
  ibrav = 0,
  a = 3.71,
  nat = 4,
  ntyp = 2,
  ecutwfc = 25.0,
  occupations = tetrahedra,
! occupations = 'smearing'
! smearing = 'mp'
! degauss = 0.01
/
&electrons
  mixing_beta = 0.7,
  conv_thr = 0.01,
/
CELL_PARAMETERS {alat}
0.5000000000000000 -0.8660254038000000 0.0000000000000000
0.5000000000000000 0.8660254038000000 0.0000000000000000
0.0000000000000000 0.0000000000000000 1.5336927200000000
ATOMIC_SPECIES
Mn 54.90000 Mn_PSEUDO
As 74.90000 As_PSEUDO
ATOMIC_POSITIONS {crystal}
Mn 0.0000000000000000 0.0000000000000000 0.0000000000000000
Mn 0.0000000000000000 0.0000000000000000 0.5000000000000000
As 0.3333333333000000 0.6666666667000000 0.2500000000000000
As 0.6666666667000000 0.3333333333000000 0.7500000000000000
K_POINTS (automatic)
6 6 6 0 0 0
```

alat = sqrt(v1*v1)

bohr

angstrom

CELL_PARAMETERS { alat | bohr | **angstrom }**

v1(1) v1(2) v1(3) <-- 1st lattice vector

v2(1) v2(2) v2(3) <-- 2nd

v3(1) v3(2) v3(3) <-- 3rd

```
&system
```

```
ibrav = 0,
```

```
a = 3.71,
```

```
nat = 4,
```

```
ntyp = 2,
```

```
ecutwfc = 25.0,
```

```
! ecutrho = 100.0,
```

```
occupations = tetrahedra,
```

```
! occupations = 'smearing'
```

```
! smearing = 'mp'
```

```
! degauss = 0.01
```

```
/
```

カットオフエネルギー

Ry 単位

自動で $\text{ecutrho} = \text{ecutwfc} * 4.0$

USPP のときは 4.0 より大きめに

書き換えるところ (本質的に自動化しにくい場所+自動化出来ない場所)

Pseudo Potential 情報

```
&control  
  calculation = 'scf',  
  prefix = 'input',  
  pseudo_dir = './',  
  outdir = './work/',  
/  
  
ATOMIC_SPECIES  
Mn 54.90000 Mn_PSEUDO  
As 74.90000 As_PSEUDO
```

必要な PP ファイルをコピーする場所



コピーしたPP名



擬ポテンシャル

PWScf で扱える擬ポテンシャルのフォーマット

CPMD Fritz Haber Unified Pseudopotential Format



USPP

<http://www.physics.rutgers.edu/gbrv/>

Pseudopotentials

Here you can download some pseudopotentials I've generated for Quantum-ESPRESSO and QE-GIPAW. **Important:** always test them on simple systems before doing any serious calculation!

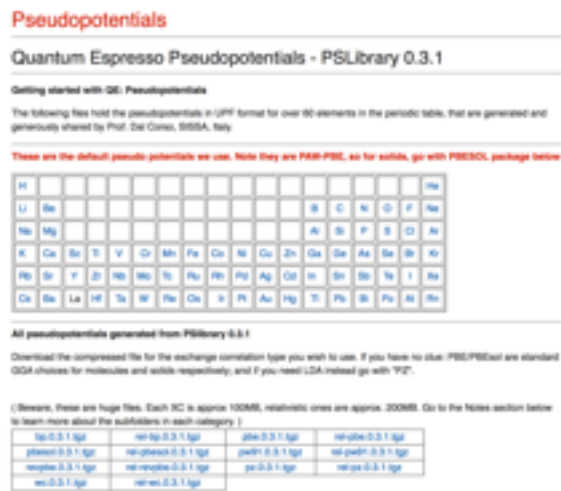
[illegible]

Nomenclature

- *sl*: *sl*, *slgs*, *psw1*, *pbe*: XC functional
- *trc*: norm-conserving, Martins Troullier method
- *nkl*: norm-conserving, RKXJ method
- *nklus*: ultrasoft, RKXJ method
- *paw*: PAW
- *gipaw*: with GIPAW reconstruction
- *semc*: with semicore orbitals
- *slc*: generated with *ldl.x* by Davide Ceresoli; *xy* by Xue Yong
- *sh*: generated with Nathalie Holzwarth's *atompaw*

If quantum-Espresso issues an error while reading GIPAW pseudopotentials, edit the UPF file, search for the `PP_PAW_FORMAT_VERSION` and `PP_GIPAW_FORMAT_VERSION` and change the content from "0.1" to "1.0".

core-hole の計算をするときには core-hole
が可能なGIPAW形式が必要となる



PSLibrary にある Quantum Espresso の PAW-PP

<http://theosrv1.epfl.ch/Main/Pseudopotentials>

PSEUDOPOTENTIAL (PAW and US) 安心

widely tested and no error has been reported so far.

PSEUDOPOTENTIAL_TOT 微妙

tested in fewer electronic environments and in some cases (mostly in fully relativistic form) encountered an error.

PSEUDOPOTENTIAL_ALT

alternative data sets that have a different number of valence states with respect to the data set in the main distribution files in the PSEUDOPOTENTIAL or PSEUDOPOTENTIAL_TOT directories. Verification tests in a wide range of electronic environments have not been performed yet.

PSEUDOPOTENTIAL_NC 危険

are the norm conserving pseudopotentials. Not tested.

PWScf における主なPPファイル名ルール

- ae: All-Electron (no pseudization)
- mt: Martins-Troullier
- bhs: Bachelet-Hamann-Schlueter and derived
- vbc: Von Barth-Car (direct fit)
- van: Vanderbilt ultrasoft
- rrkj: Rappe Rabe Kaxiras Joannopoulos (norm-conserving)
- rrkjus: Rappe Rabe Kaxiras Joannopoulos (ultrasoft)
- kjpaw: Projector Augmented Wave (Kresse-Joubert paper)
- bpaw: Projector Augmented Wave (original Bloechl recipe)

full-relativistic

Si.**rel**-**pbe**-**n**-**kjpaw**_psl.0.1.UPF

- pz: Perdew-Zunger (LDA)
- vwn: Vosko-Vilk-Nusair (LDA)
- pbe: Perdew-Burke-Ernzerhof (PBE)
- blyp: Becke-Lee-Yang-Parr (BLYP)
- pw91: Perdew-Wang 91 gradient-corrected functional
- tpss: Tao-Perdew-Staroverov-Scuseria (TPSS) meta-GGA
- coulomb: Coulomb teta-Z/r potential

- s: semicore state s in valence
- p: semicore state p in valence
- d: semicore state d in valence
- f: valence f state in core
- n: nonlinear core-correction

XANES などの計算が前提の時は

1) **PWmakePPinp** Cu.pbe-dn-kjpaw_psl.0.2.UPF > **Cu.in**

2) **Cu.in** の修正

3) **ld1.x** < **Cu.in** > **output.ld1.out**

SCF計算 (基本)

実行(single)

```
pw.x < scf.in > out
```

実行(mpi)

```
mpirun -np 16 pw.x < scf.in > out
```

注意) タグの幾つかはかならずバインドされる
(デフォルト値は存在しない)ので必須となる

```
! occupations = tetrahedra,  
occupations = 'smearing',  
! smearing = 'mp'  
smearing = 'gaussian'  
degauss = 0.01
```

```
nspin = 2,  
starting_magnetization(4) = 0.3
```

```
/
```

smearing を指定した時は
smearing の方法論の設定が必須
gaussian のときはdegauss が必要となる

nspin を導入した時は必ず startingが必要

注意) ルールを守らないと意味不明のエラーで止まる

文字列はシングルクォートで囲む”
数値はそのまま

```
occupations = 'smearing',  
smearing = 'gaussian'  
degauss = 0.01  
nspin = 2,  
starting_magnetization(4) = 0.3
```

SPIN分極計算

&control

calculation = 'scf',

prefix = 'input',

pseudo_dir = './',

outdir = './work/',

nspin=2,

starting_magnetization(1)= 0.6



nscf モードの時は**除く**
(**scf** モードの時は**必須**)

分子系の計算メモ

分子系の計算(Γ点計算)

K_POINTS を **automatic** のまま

1 1 1 0 0 0

などに**しない**ほうがよい

K_POINTS {gamma}

ただし、XSpectra では 1 1 1 0 0 0 とする

DOS計算

DOS 計算

1) K-meshを増やした SCF計算

tetrahedron

nscf モード

2) DOS 計算

&control

calculation = 'nscf',

prefix = 'input',

pseudo_dir = './',

outdir = './work/',

/

&system

ibrav = 0,

a = 3.71,

nat = 4,

ntyp = 2,

ecutwfc = 25.0,

occupations = tetrahedra,

/

&electrons

mixing_beta = 0.7,

conv_thr = 0.01,

/

CELL_PARAMETERS {alat}

0.5000000000000000 -0.866025403800000 0.0000000000000000

0.5000000000000000 0.866025403800000 0.0000000000000000

0.0000000000000000 0.0000000000000000 1.5336927200000000

ATOMIC_SPECIES

Mn 54.90000 Mn_PSEUDO

As 74.90000 As_PSEUDO

ATOMIC_POSITIONS {crystal}

Mn 0.0000000000000000 0.0000000000000000 0.0000000000000000

Mn 0.0000000000000000 0.0000000000000000 0.5000000000000000

As 0.3333333333300000 0.666666666700000 0.2500000000000000

As 0.666666666700000 0.333333333300000 0.7500000000000000

K POINTS (automatic)

12 12 12 0 0 0

DOS用のSCF計算のコントロールをチェック

```
&control  
  calculation = 'nscf',  
  prefix = 'input',  
  pseudo_dir = './',  
  outdir = './work/',  
/
```



```
&dos  
  outdir = './work/',  
  prefix='input',  
  fildos='MnAs.dos',  
/
```

dos.x < dos.in > dos.out



GNUPLOT 形式

energy DOS TDOS

部分状態密度の計算 (projwfc.x)

&projwfc

outdir = './work/',

prefix='input',

filpdos='MnAs.pdos',

/

projwfc.x < pdos.in > pdos.out

MnAs.dos

MnAs.pdos.pdos_atm#1 (Mn)_wfc#1 (s)

MnAs.pdos.pdos_atm#1 (Mn)_wfc#2 (s)

MnAs.pdos.pdos_atm#1 (Mn)_wfc#3 (p)

MnAs.pdos.pdos_atm#1 (Mn)_wfc#4 (d)

site1

MnAs.pdos.pdos_atm#2 (Mn)_wfc#1 (s)

MnAs.pdos.pdos_atm#2 (Mn)_wfc#2 (s)

MnAs.pdos.pdos_atm#2 (Mn)_wfc#3 (p)

MnAs.pdos.pdos_atm#2 (Mn)_wfc#4 (d)

site2

MnAs.pdos.pdos_atm#3 (As)_wfc#1 (s)

MnAs.pdos.pdos_atm#3 (As)_wfc#2 (p)

site3

MnAs.pdos.pdos_atm#4 (As)_wfc#1 (s)

MnAs.pdos.pdos_atm#4 (As)_wfc#2 (p)

site4

MnAs.pdos.pdos_tot

ATOMIC_POSITIONS {crystal}

Mn	0.0000000000	0.0000000000	0.00
----	--------------	--------------	------

Mn	0.0000000000	0.0000000000	0.50
----	--------------	--------------	------

As	0.3333333333	0.6666666667	0.25
----	--------------	--------------	------

As	0.6666666667	0.3333333333	0.75
----	--------------	--------------	------

LDOS と PDOS がそれぞれ出力される

# E (eV)	ldos(E)	pdos(E)	pdos(E)	pdos(E)
-71.129	0.264E-08	0.745E-09	0.946E-09	0.946E-09
-71.119	0.645E-06	0.169E-06	0.238E-06	0.238E-06
-71.109	0.252E-04	0.577E-05	0.973E-05	0.973E-05

PDOS計算の注意

&projwfc

outdir = './work/',

prefix='input',

filpdos='input.pdos',

ngauss = 0, ← Gaussian ブロードニング

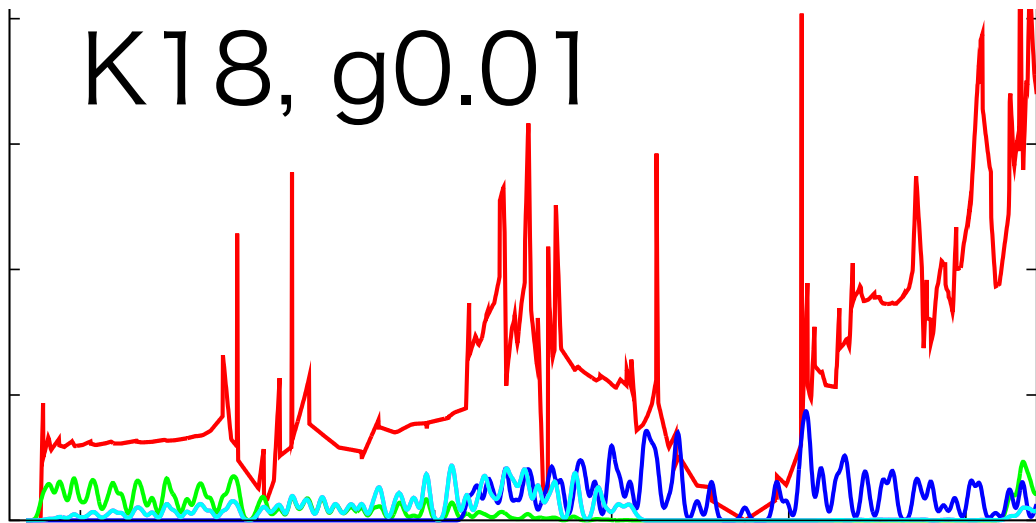
degauss = 0.02,

DeltaE = 0.01,

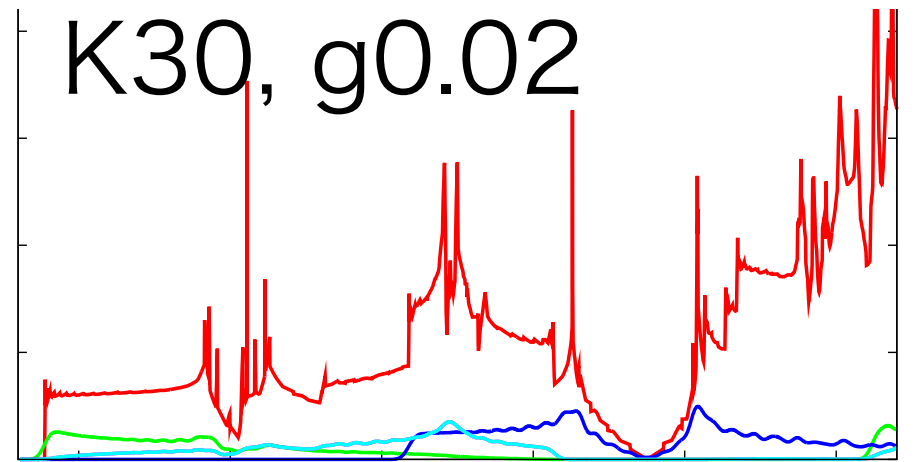
/

ブロードニングのSigma値を 0.02[eV]あたりにして、
K-Mesh を 30x30x30 くらいにしないとまったくPDOS
が役に立たない

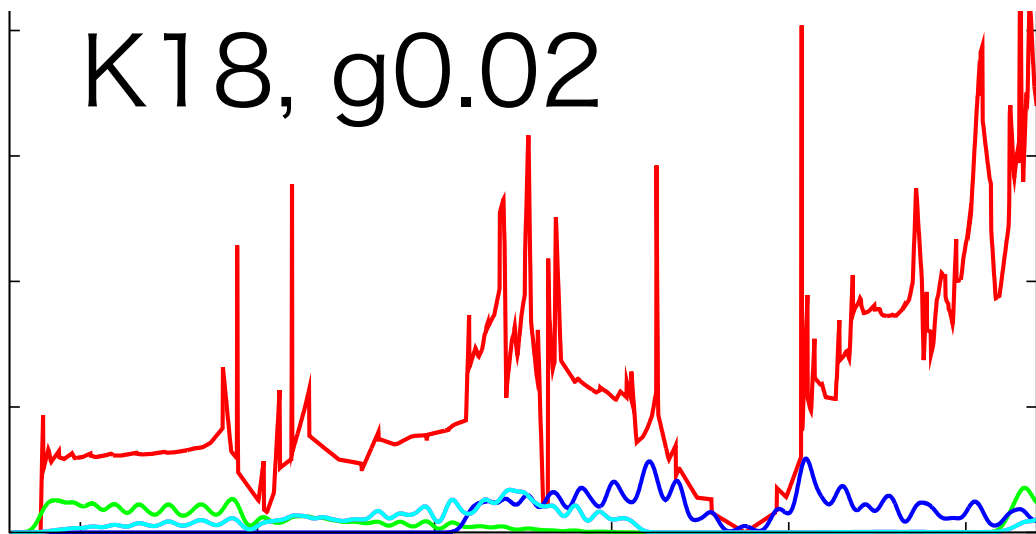
K18, g0.01



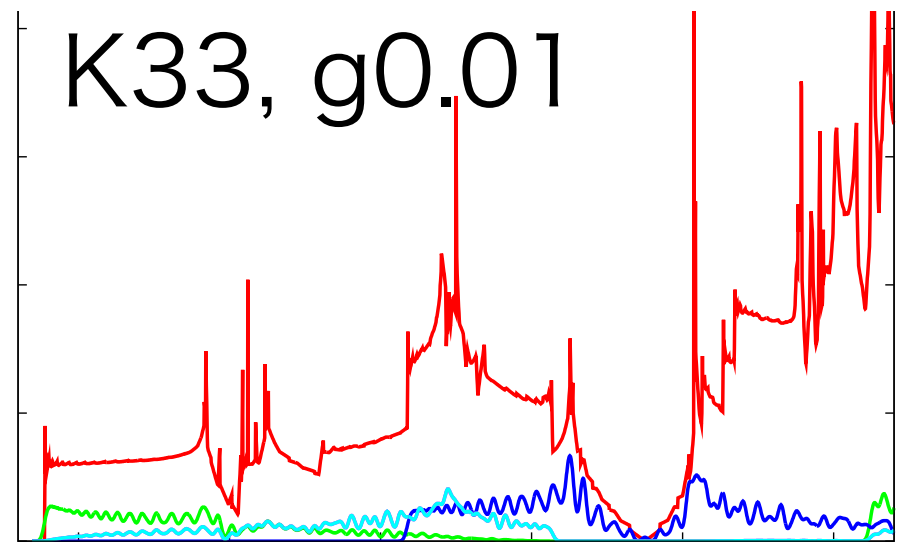
K30, g0.02



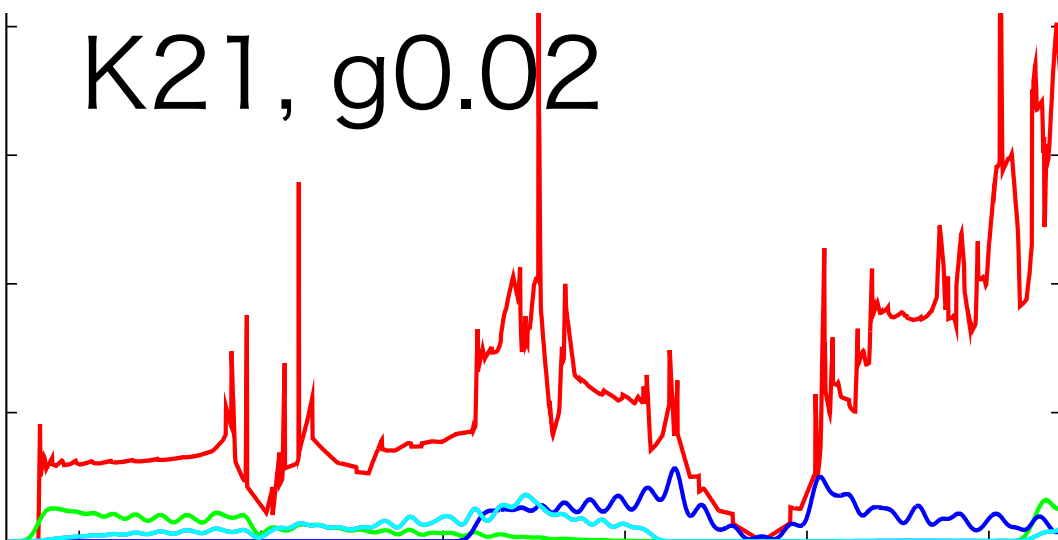
K18, g0.02



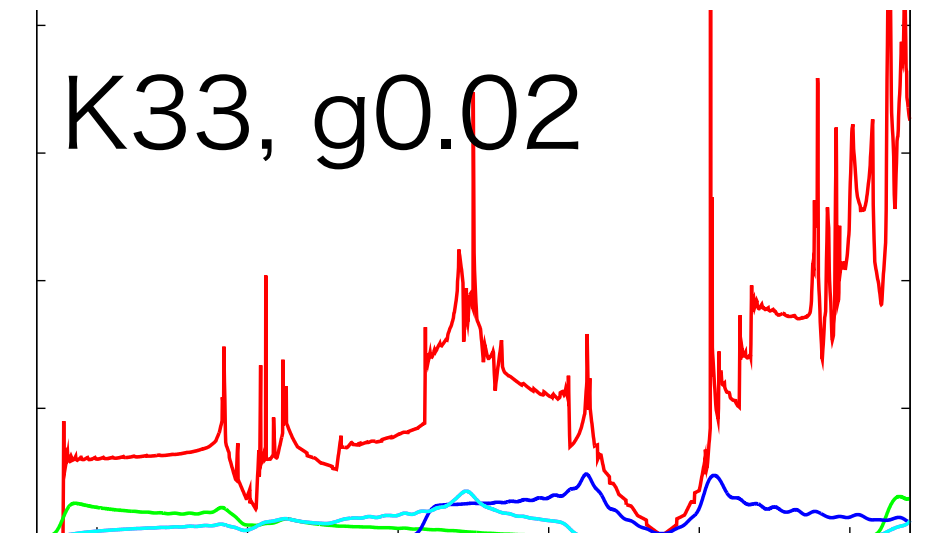
K33, g0.01

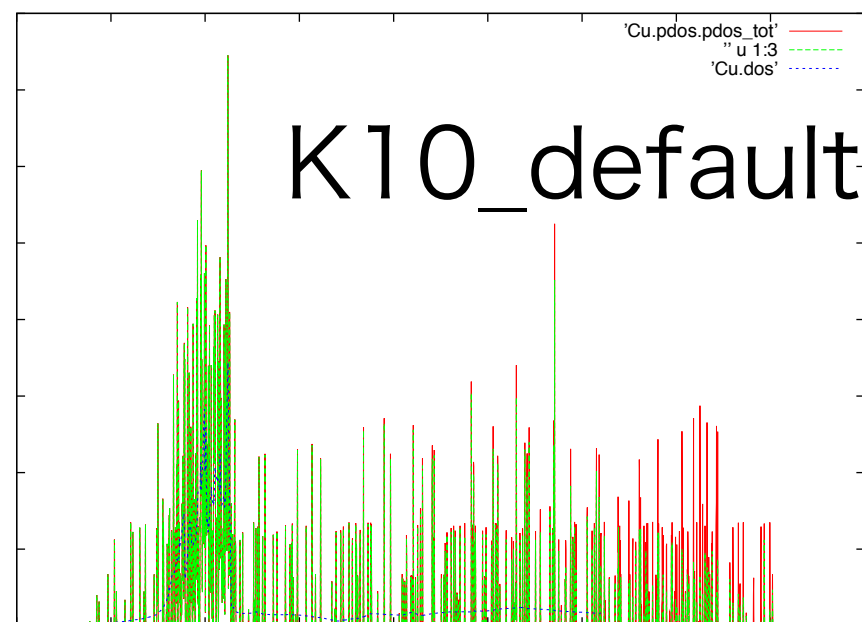
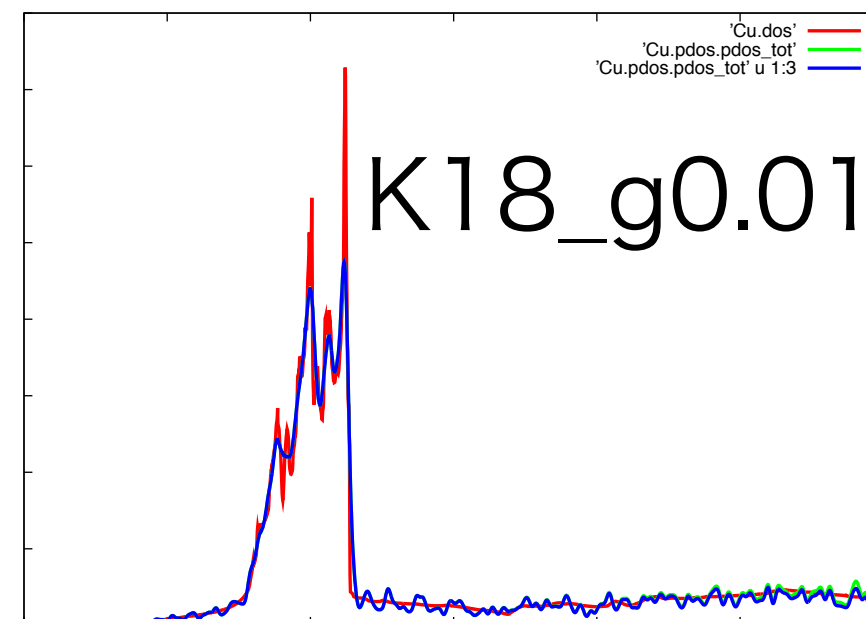
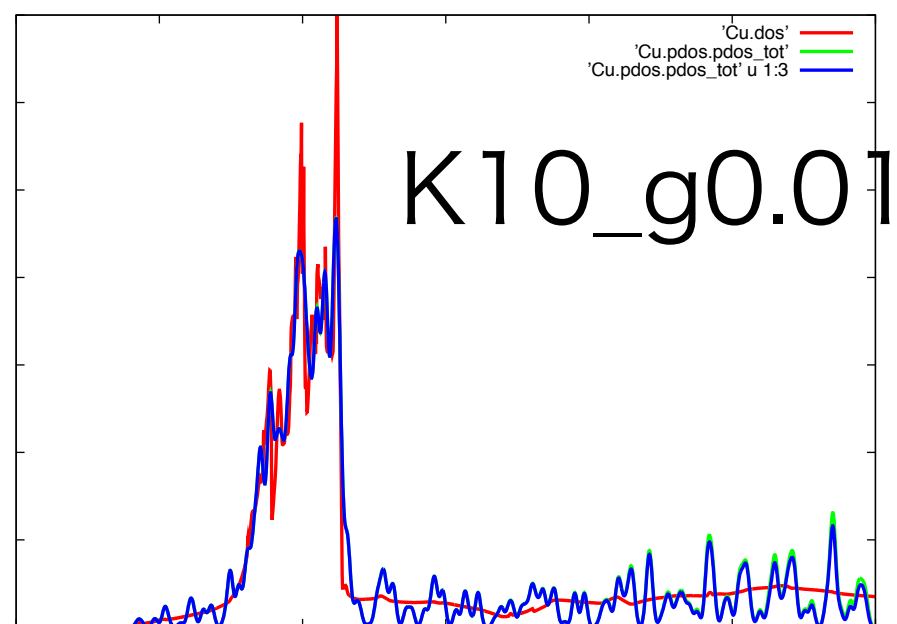
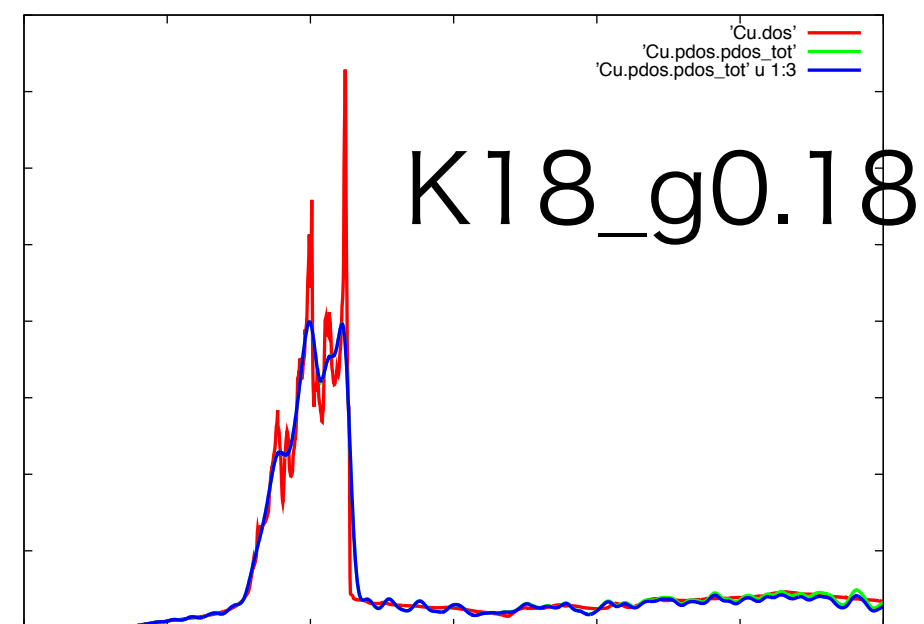
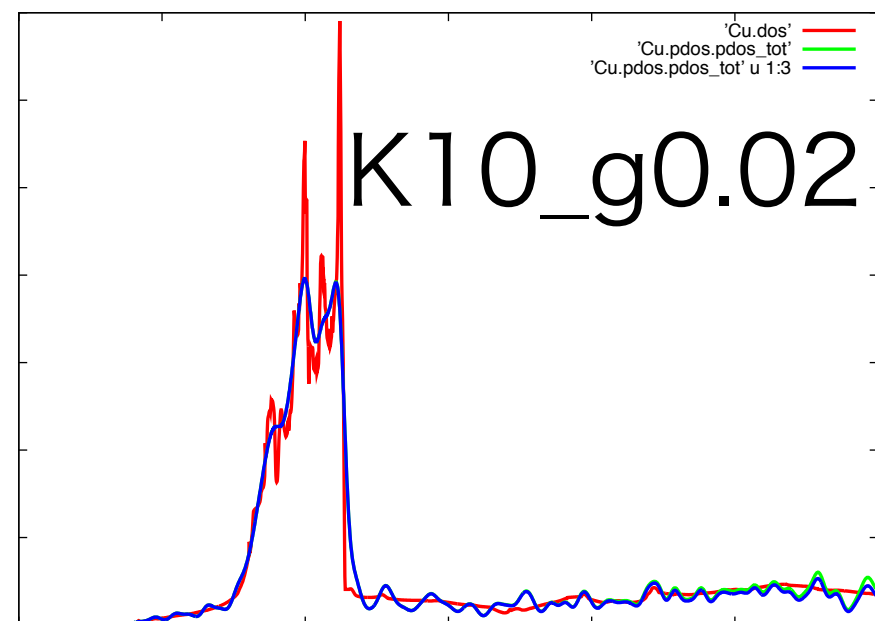
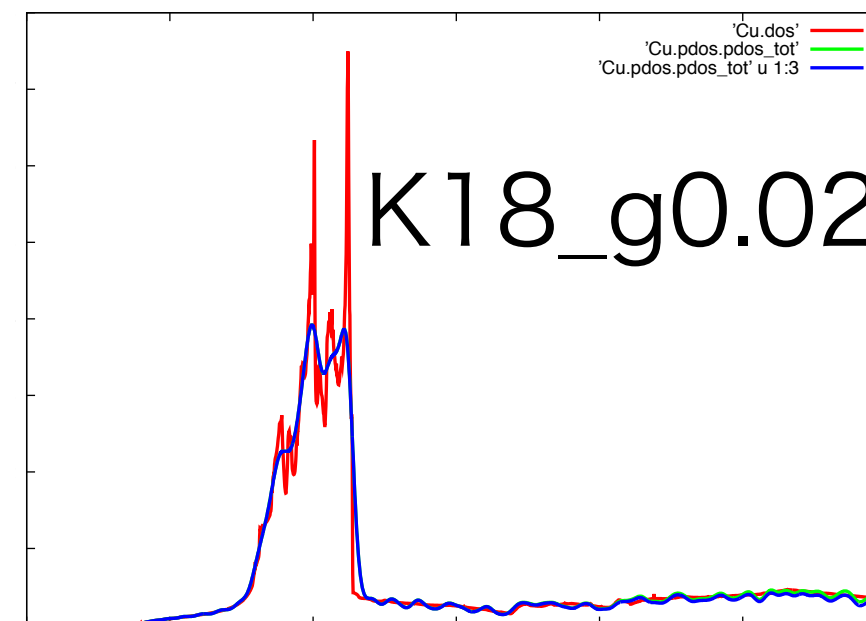


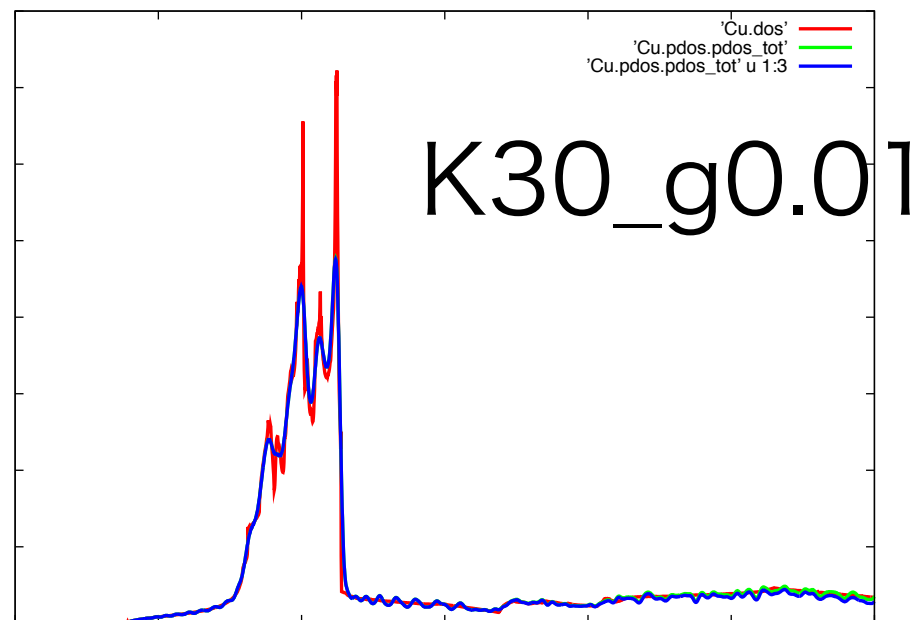
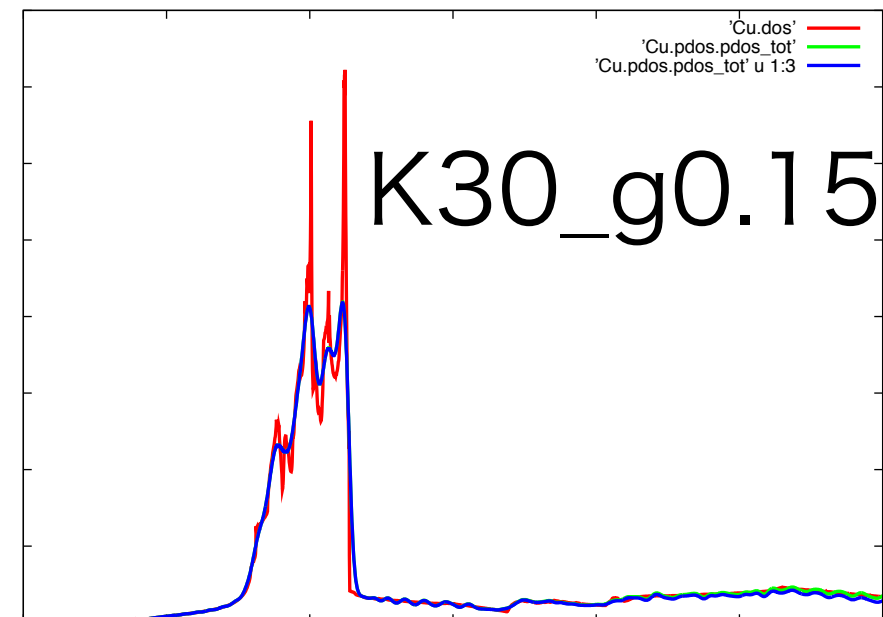
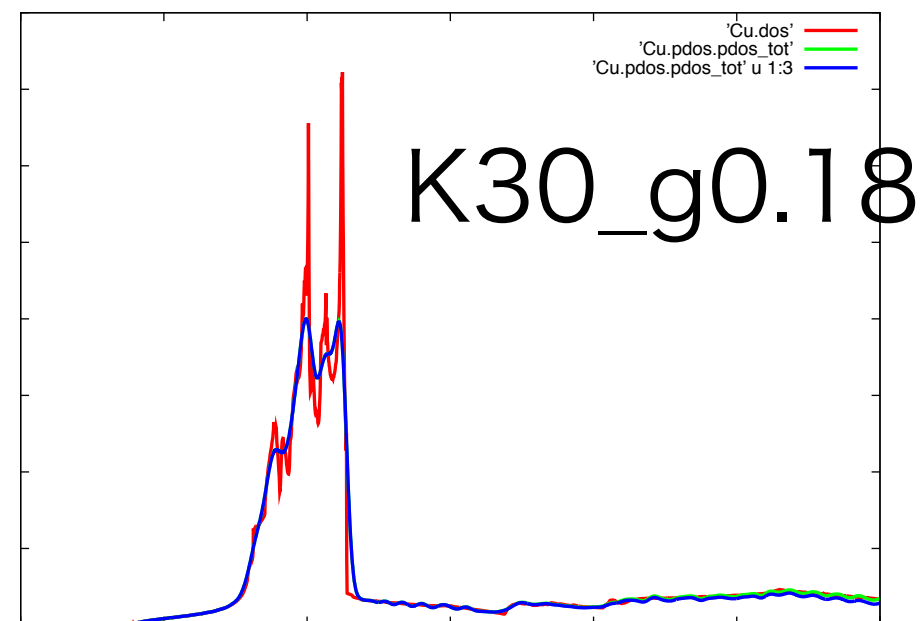
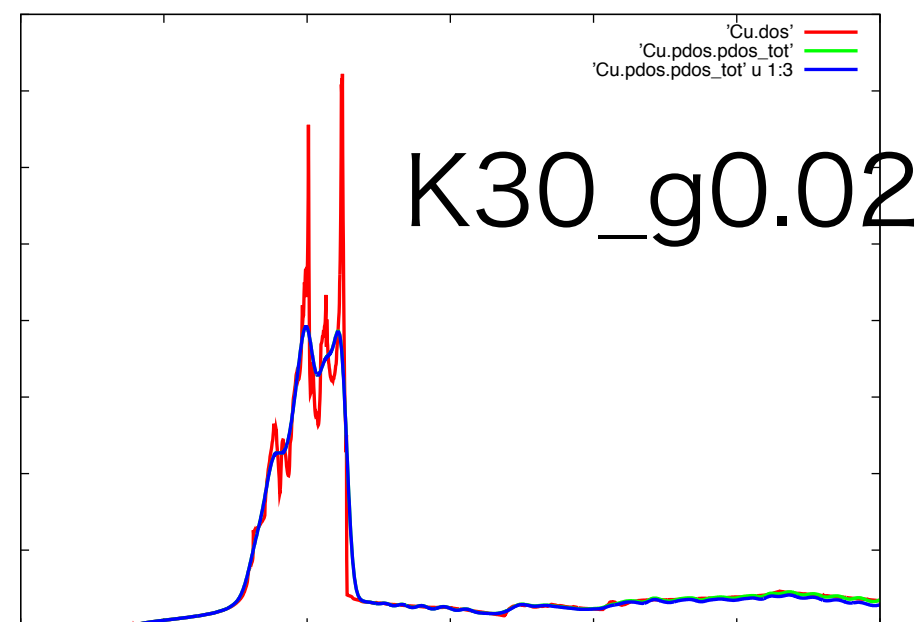
K21, g0.02



K33, g0.02







XAS

ld1.x

xspectra.x

(XSPECTRA: K-edge X-ray adsorption spectra)

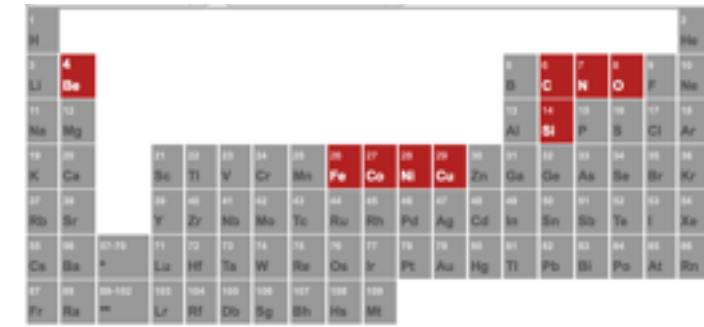
K-edge(1s) と L1-edge(2s) が可能

始状態のコア波動関数作成

始状態は必ず full-core (かならず必要)

a) quantum espresso で pseudopotential が公開されている可能性

<http://www.quantum-espresso.org/pseudopotentials/>



A periodic table of elements where the elements Copper (Cu), Iron (Fe), Nickel (Ni), and Cobalt (Co) are highlighted in red, indicating the availability of pseudopotentials for these elements.

b) **GIPAW_CORE_POTENTIAL** から **upf2plotcore.sh** で波動関数の吸い出し

GIPAW ポテンシャルファイルが必要

<https://sites.google.com/site/dceresoli/pseudopotentials>

例)

Cu.pbe-kjpaw.UPF ではダメで

Cu.pbe-dn-kjpaw_psl.0.2.UPF もダメ

Cu.pbe-n-van_gipaw.UPF が必要

Pseudopotentials

Here you can download some pseudopotentials I've generated for Quantum-ESPRESSO and QE-GIPAW
Important: always test them on simple systems before doing any serious calculation!



A periodic table of elements where the elements from Lithium (Li) to Xenon (Xe) are highlighted in yellow, and the elements from Lanthanum (La) to Oganesson (Og) are highlighted in green, indicating the availability of pseudopotentials for these elements.

upf2plotcore.sh **Cu.pbe-n-van_gipaw.UPF** > **Cu.wfc**

GIPAW ポテンシャル

コア波動関数

core-hole ポテンシャル&波動関数

1 H																	2 He	
3 Li	4 Be							5 B	6 C	7 N	8 O	9 F	10 Ne					
11 Na	12 Mg							13 Al	14 Si	15 P	16 S	17 Cl	18 Ar					
19 K	20 Ca		21 Sc	22 Ti	23 V	24 Cr	25 Mn	26 Fe	27 Co	28 Ni	29 Cu	30 Zn	31 Ga	32 Ge	33 As	34 Se	35 Br	36 Kr
37 Rb	38 Sr		39 Y	40 Zr	41 Nb	42 Mo	43 Tc	44 Ru	45 Rh	46 Pd	47 Ag	48 Cd	49 In	50 Sn	51 Sb	52 Te	53 I	54 Xe
55 Cs	56 Ba	57-70 *	71 Lu	72 Hf	73 Ta	74 W	75 Re	76 Os	77 Ir	78 Pt	79 Au	80 Hg	81 Tl	82 Pb	83 Bi	84 Po	85 At	86 Rn
87 Fr	88 Ra	89-102 **	103 Lr	104 Rf	105 Db	106 Sg	107 Bh	108 Hs	109 Mt									

GIPAW ポテンシャル

Pseudopotentials

Here you can download some pseudopotentials I've generated for Quantum-ESPRESSO and QE-GIPAW.
Important: always test them on simple systems before doing any serious calculation!

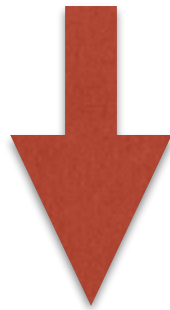
1 H																	2 He
3 Li	4 Be											5 B	6 C	7 N	8 O	9 F	10 Ne
11 Na	12 Mg											13 Al	14 Si	15 P	16 S	17 Cl	18 Ar
19 K	20 Ca	21 Sc	22 Ti	23 V	24 Cr	25 Mn	26 Fe	27 Co	28 Ni	29 Cu	30 Zn	31 Ga	32 Ge	33 As	34 Se	35 Br	36 Kr
37 Rb	38 Sr	39 Y	40 Zr	41 Nb	42 Mo	43 Tc	44 Ru	45 Rh	46 Pd	47 Ag	48 Cd	49 In	50 Sn	51 Sb	52 Te	53 I	54 Xe
55 Cs	56 Ba	71 Lu	72 Hf	73 Ta	74 W	75 Re	76 Os	77 Ir	78 Pt	79 Au	80 Hg	81 Tl	82 Pb	83 Bi	84 Po	85 At	86 Rn
87 Fr	88 Ra	103 Lr	104 Rf														
		57 La	58 Ce	59 Pr	60 Nd	61 Pm	62 Sm	63 Eu	64 Gd	65 Tb	66 Dy	67 Ho	68 Er	69 Tm	70 Yb		
		89 Ac	90 Th	91 Pa	92 U	93 Np	94 Pu	95 Am	96 Cm	97 Bk	98 Cf	99 Es	100 Fm	101 Md	102 No		

<https://sites.google.com/site/dceresoli/pseudopotentials>

GIPAW 等で コア状態の波動関数を含めた擬ポテンシャルが公開されていない場合

ld1.x で計算

ld1.x < **input.ld1.in** > output.ld1.out



ld1.x 入力ファイルの作成 (input.ld1.in)

擬ポテンシャルファイル中の<PP_INPUTFILE> </PP_INPUTFILE>
の間の部分を入力ファイルとして取り出す。

拙作スクリプト) **PWmakePPinp**

1) **PWmakePPinp** Cu.pbe-dn-kjpaw_psl.0.2.UPF > Cu.in

例) Au

0) ld1.x 入力ファイルの作成

PWmakePPinp.py Au.pbe-dn-kjpaw_psl.0.1.UPF
> Au.in

1) GIPAW を再構成するオプション追加

追加オプション

lgipaw_reconstruction = .true.

2) ld1.x の実行

新しい UPF ファイルが作成される

3) コアの wfc ファイルを作成

upf2plotcore.sh *.UPF> Au.wfc

```
&input
  title='Au',
  zed=79.,
  rel=1,
  config='[Xe] 4f14 5d10.0 6s1.0 6p0.0',
  iswitch=3,
  dft='PBE'
/
&inputp
  lpaw=.true.,
  pseudotype=3,
  file_pseudopw='Au.pbe-dn-kjpaw_psl.0.1+.UPF',
  author='ADC',
  lloc=-2,
  rcloc=2.4,
  which_augfun='PSQ',
  rmatch_augfun=0.95,
  nlcc=.true.,
  new_core_ps=.true.,
  rcore=1.6,
  tm=.true.
lgipaw_reconstruction = .true.
/
6
6S 1 0 1.00 0.00 1.90 2.30 0.0
6S 1 0 0.00 2.30 1.70 2.30 0.0
6P 2 1 0.00 0.00 2.00 2.40 0.0
6P 2 1 0.00 3.60 1.80 2.40 0.0
5D 3 2 10.00 0.00 1.80 2.20 0.0
5D 3 2 0.00 -0.30 1.10 2.20 0.0
```

★名前変更★

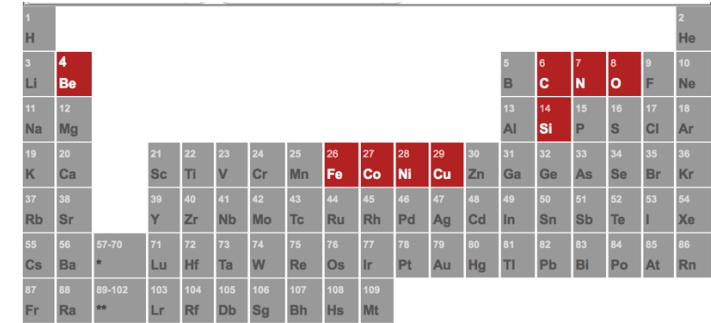
★追加★

終状態のコア波動関数作成

終状態のwcf(コア波動関数)を用意する(with core-hole or without core-hole)

a) quantum espresso で pseudopotential が公開されている可能性

<http://www.quantum-espresso.org/pseudopotentials/>

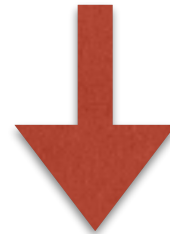


A periodic table of elements. Elements highlighted in red include: Be (4), C (6), N (7), O (8), Si (14), P (15), S (16), Cl (17), Ar (18), Fe (26), Co (27), Ni (28), Cu (29), Zn (30), Ga (31), Ge (32), As (33), Se (34), Br (35), Kr (36), Y (39), Zr (40), Nb (41), Mo (42), Tc (43), Ru (44), Rh (45), Pd (46), Ag (47), Cd (48), In (49), Sn (50), Sb (51), Te (52), I (53), Xe (54), Lu (71), Hf (72), Ta (73), W (74), Re (75), Os (76), Ir (77), Pt (78), Au (79), Hg (80), Tl (81), Pb (82), Bi (83), Po (84), At (85), Rn (86), and the lanthanide and actinide series at the bottom.

b) ld1.x で作成

ld1.x で計算

ld1.x < **input.ld1.in** > output.ld1.out



ld1.x 入力ファイルの作成 (input.ld1.in)

擬ポテンシャルファイル中の<PP_INPUTFILE> </PP_INPUTFILE>
の間の部分を入力ファイルとして取り出す。

拙作スクリプト) **PWmakePPinp**

1) **PPmakePPinp** Cu.pbe-dn-kjpaw_psl.0.2.UPF > Cu.in

コアホールを開けるときは ld1.x の入力ファイルに注意

config の修正

基本は config の修正だけでOK

&input

title='Cu',
zed=29.,
rel=1,

config='[Ar] 3d9.5 4s1.5 4p0.0',

iswitch=3,
dft='PBE'

/

&inputp

lpaw=.true.,
pseudotype=3,
file_pseudopw='Cu.pbe-dn-kjpaw_psl.0.2.UPF',
author='ADC',
lloc=-1,
rcloc=1.9,
which_augfun='BESSEL',
rmatch_augfun=1.8,
nlcc=.true.,
rcore=0.7,
tm=.true.

lgipaw_reconstruction = .true.

/

6
4S 1 0 1.50 0.00 2.20 2.30 0.0
4S 1 0 0.00 6.10 1.40 2.30 0.0
4P 2 1 0.00 0.00 2.20 2.30 0.0
4P 2 1 0.00 6.30 1.40 2.30 0.0
3D 3 2 9.50 0.00 1.50 1.90 0.0
3D 3 2 0.00 2.30 1.50 1.90 0.0

config の修正

基本は config の修正だけでOK

zval があるときは hole
にしたぶん電子を抜く

configu 中に [Ar] とあるものは Ar コ
アなので省略形は止めてすべて書き下す

Ar = 1s2 2s2 2p6 3s2 3p6

config =

1s2 2s2 2p6 3s2 3p6 3d9.5 4s1.5 4p0.0

1sに core-hole

config =

1s1 2s2 2p6 3s2 3p6 3d9.5 4s1.5 4p0.0

擬ポテンシャル生成スイッチ

作成する擬ポテン
シャルの名前

★追加★

xanes_dipole or xanes_quadrupole

```
&input_xspectra
```

```
  calculation='xanes_dipole',
```

```
  prefix='pwscf',
```

```
  outdir='work',
```

```
  xonly_plot=.false.,
```

```
  xniter=1000,
```

```
  xcheck_conv=50,
```

```
  xerror=0.001,
```

Perform a full continuous
fraction calculation

control the Lanczos

Lanczos method の系数
を保存するファイル

```
  x_save_file='prefix.xspectra.sav',
```

```
  ef_r=0.574496,
```

★フェルミレベル(Ry単位)★

```
  xiabs=3,
```

```
  xepsilon(1)=1.0,
```

```
  xepsilon(2)=0.0,
```

```
  xepsilon(3)=0.0,
```

吸収原子の選択

★ATOMIC_SPECIES in pwscf input)★

ビームによる電場方向

/

```

/
&plot      ★(orentzian broadening, related to the core-hole lifetime )★
  xgamma=0.7, linewidth to be used in the spectrum [eV]

  xnepoint=1000, number of energy points in the plot of the XAS spectrum

  xemin=-10.0,
  xemax=30.0,

  terminator=.true., to use the terminator function for the continued fraction
  cut_occ_states=.true.,
/          the occupied states are smoothly cut from the plot

&pseudos
  filecore='Cu.wfc', ★吸収端★のcore状態(始状態) の波動関数
  ! r_paw(1)=3.2,
/

&cut_occ
  cut_desmooth=0.1,
  cut_step1=0.01,
/

8 8 8 0 0 0      SCF計算で用いた K-point 情報

```

&input_xspectra

■ variables

- calculation : '**xanes_dipole**', Perform dipolar calculation. '**xanes_quadrupole**', Perform quadrupolar calculation.
- xonly_plot : (default .false.) .false. the continued fraction is calculated for each k-point and at the end written on the save file.
- xniter : (default 2000) maximum number of iterations for lanczos.
- xcheck_conv : (default 50) number of iterations between 2 convergence tests
- xerror : (default 0.01) convergence threshold for lanczos calculation (eV)
- **xiabs** : type of the absorbing atom, (ATOMIC_SPECIES in pwscf input)
- xepsilon : (default 1.0 0.0 0.0) coordinates of the incident x-ray polarization vector
- ef_r : **Fermi energy in Rydberg**. This value combined with the option cut_occupied_states can be used to exclude the occupied states in a smooth way from the final plot.
- xgamma : (default 0.1) **linewidth to be used in the spectrum (eV)**
- xnepoint : (default 1000) number of energy points in the plot of the XAS spectrum
- **xemin** : (default 0.0) minimum energy in eV for the plot of the XAS spectrum
- **xemax** : (default 10.0) maximum energy in eV for the plot of the XAS spectrum
- terminator : (default .false.) .false. no terminator is used.
- cut_occ_states : (default .false.) .false. the occupied states are visualized
- filecore : core wavefunction file
- r_paw(1) : (default 1.5*rc) paw radii to be used in paw reconstruction. A good choice to avoid linearly dependent projectors is 3*r_pseudo/2 or grater.
- cut_desmooth : (default 0.01) size of the interval near the fermi energy in which cross section is smoothed
- cut_stepl : (default 0.01) integration initial step, lower side

始状態(nch (non-core-hole))

1) GIPAW-PP ポテンシャルファイルの作成

A) **PWmakePPinp** Cu.pbe-dn-kjpaw_psl.0.2.UPF > Cu.in

B) vi Cu.in

lgipaw_reconstruction = .true.

file_pseudopw=***

C) **ld1.x** < Cu.in > ld1.output

4) 作成した新たな UPF ファイル (GIPAW-PPポテンシャル) を用いて WFC を作成

A) **upf2plotcore.sh *.UPF** > Cu.wfc

終状態(full-core state)

Ne: 1s(2) 2s(2) 2p(6)

Ar: [Ne] 3s(2) 3p(6)

Kr: [Ar] 3d(10) 4s(2) 4p(6)

Xe: [Kr] 4d(10) 5s(2) 5p(6)

Xe-core

1s² 2s² 2p⁶ 3s² 3p⁶ 3d¹⁰ 4s² 4p⁶ 4d¹⁰ 5s² 5p⁶

Au原子

[Xe] 4f¹⁴ 5d¹⁰ 6s¹

すべて書き下すと

1s² 2s² 2p⁶ 3s² 3p⁶ 3d¹⁰ 4s² 4p⁶ 4d¹⁰ 5s² 5p⁶ 4f¹⁴ 5d¹⁰ 6s¹

Au L1-fch (2s軌道に full-core-holeをひとつ作る)

config='1s2 **2s1** 2p6 3s2 3p6 3d10 4s2 4p6 4d10 5s2 5p6 4f14 5d10.0 6s1.0 6p0.0

pw.x SCF用のinput ファイルで設定

scf 計算の時に忘れずに

tot_charge = +1.0

計算する固有値を増やす

1x1 Graphene の valence の電子数 (Cだと4個(both-spin))

ログ出力

number of atoms/cell	=	2
number of electrons	=	8.00

nbnd は計算されるスピン毎のバンド数 (の電子数)

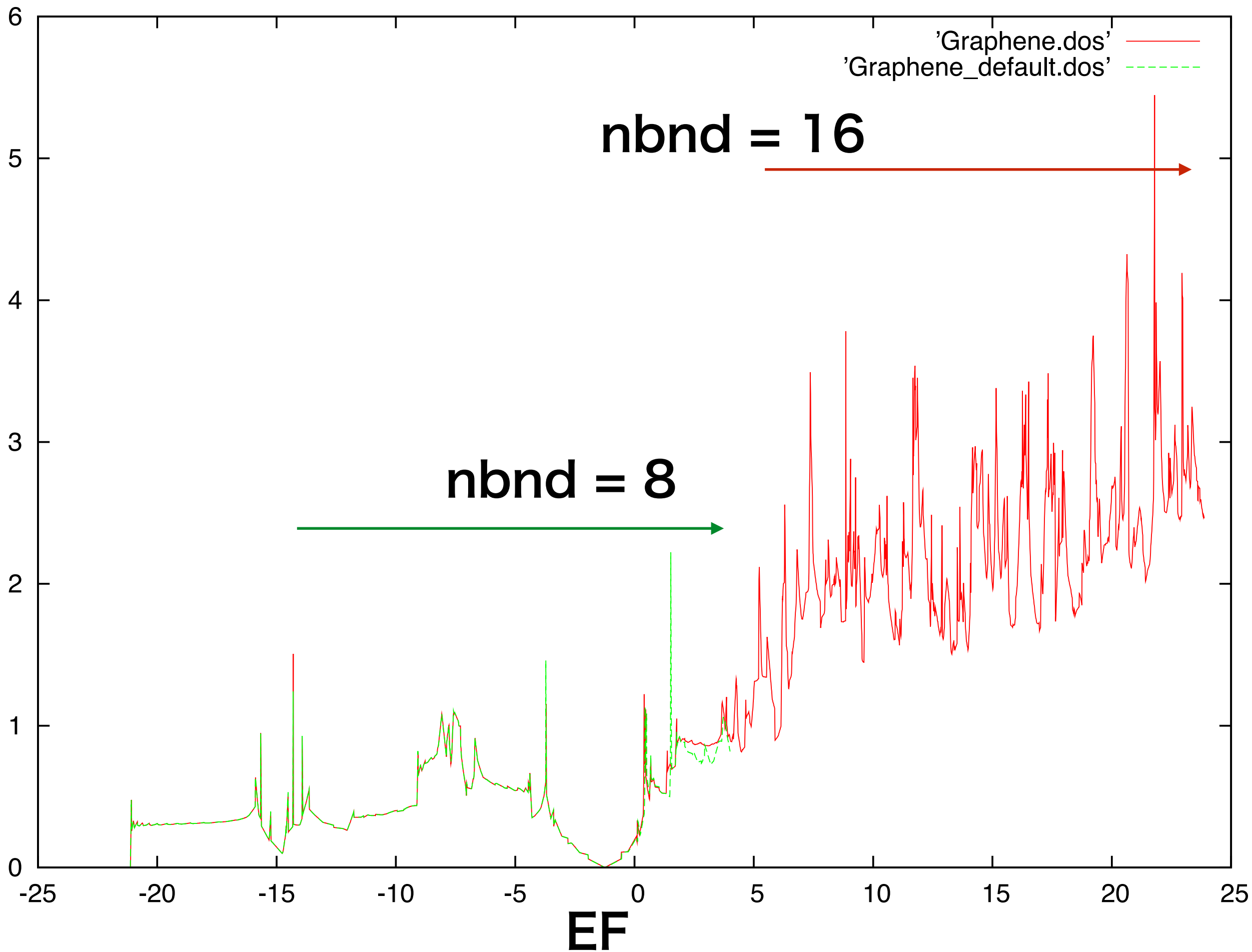
default 値 (この場合) nbnd = 8 となる



ログ出力

number of Kohn-Sham states= 8.00

注意) 電子数=nbnd とは限らない(nbnd はスピン毎なので
電子数の半分あればvalence全部のスピンバンドをカバーできる)



コアホールを開けたバルクのXANES計算 まとめ

ZnO , Zn-K

ZnO の XANES の計算(テスト) の流れ

1) 始状態の計算はコア状態だけで計算可能

1-1) 始状態の擬ポテンシャルを作成

2) 終状態の計算はバンド計算が必要

2-1) 終状態の擬ポテンシャルを作成

2-2) 終状態の擬ポテンシャルを含む構造
(吸収原子とそれ意外を区別する構造) の
入力ファイルを準備する

2-3) SCF 計算

2-4) XANES計算用入力ファイルの用意

2-5) xspectra.x の実行

Zn ポテンシャルの始状態のwevefunction を作成

lgipaw_reconstruction = .true.
オプション付きで最初から作られている
擬ポテンシャルの場合は必要ないステップ

0) ld1.x 入力ファイルの作成

PWmakePPinp.py

Zn.pbe-dn-kjpaw_psl.0.2.2.UPF > Zn.in

1) GIPAW を再構成するオプション追加 追加オプション

lgipaw_reconstruction = .true.

2) ld1.x の実行

新しい UPF ファイルが作成される

3) コアの wfc ファイルを作成

upf2plotcore.sh *.UPF > Zn.wfc

```
&input
  title='Zn',
  zed=30.,
  rel=1,
  config='[Ar] 3d10. 4s2. 4p0.0',
  iswitch=3,
  dft='PBE'
/
&inputp
  lpaw=.true.,
  pseudotype=3,
  file_pseudopw='Zn.pbe-dn-kjpaw_psl.0.2.2.UPF',
  author='ADC',
  which_augfun='PSQ',
  rmatch_augfun=1.3,
  lloc=-1,
  rcloc=1.7,
  nlcc=.true.,
  rcore=0.7,
  tm=.true.
  lgipaw_reconstruction=.true.
  use_paw_as_gipaw=.true.
/
6
4S 1 0 2.00 0.00 2.10 2.20 0.0
4S 1 0 0.00 6.10 1.40 2.20 0.0
4P 2 1 0.00 0.00 2.20 2.30 0.0
4P 2 1 0.00 6.30 1.40 2.30 0.0
3D 3 2 10.00 0.00 1.50 1.80 0.0
3D 3 2 0.00 -0.60 1.50 1.80 0.0
```

★名前変更★

★追加★

Zn ポテンシャルの終状態のwevefunction を作成
full core hole を作る

ホールを開けるのならば必須

0) ld1.x 入力ファイルの作成

PWmakePPinp.py

Zn.pbe-dn-kjpaw_psl.0.2.2.UPF > **Zn_fch.in**

1) GIPAW を再構成するオプション追加

追加オプション

lgipaw_reconstruction = .true.

2) ホールを開ける

3) ld1.x の実行

4) コアの wfc ファイルを作成

&input

title='Zn',

zed=30.,

rel=1,

config='1s1 2s2 2p6 3s2 3p6 3d10. 4s2. 4p0.0',

iswitch=3,

dft='PBE'

/

&inputp

lpaw=.true.,

pseudotype=3,

★名前変更★

file_pseudopw='Zn.pbe-dn-kjpaw_psl.0.2.2_fch.U

author='ADC',

which_augfun='PSQ',

rmatch_augfun=1.3,

lloc=-1,

rcloc=1.7,

nlcc=.true.,

rcore=0.7,

tm=.true.

★追加★

lgipaw_reconstruction=.true.

use_paw_as_gipaw=.true.

/

6

4S 1 0 2.00 0.00 2.10 2.20 0.0

4S 1 0 0.00 6.10 1.40 2.20 0.0

4P 2 1 0.00 0.00 2.20 2.30 0.0

4P 2 1 0.00 6.30 1.40 2.30 0.0

3D 3 2 10.00 0.00 1.50 1.80 0.0

3D 3 2 0.00 -0.60 1.50 1.80 0.0

```

&input
  title='Zn',
  zed=30.,
  rel=1,
  config='1s1 2s2 2p6 3s2 3p6 3d10. 4s2. 4p0.0',
  iswitch=3,
  dft='PBE'
/
&inputp
  lpaw=.true.,
  pseudotype=3,
  file_pseudopw='Zn.pbe-dn-kjpaw_psl.0.2.2_fch.UPF',
  author='ADC',
  which_augfun='PSQ',
  rmatch_augfun=1.3,
  lloc=-1,
  rcloc=1.7,
  nlcc=.true.,
  rcore=0.7,
  tm=.true.
  lgipaw_reconstruction=.true.
  use_paw_as_gipaw=.true.
/
6
4S 1 0 2.00 0.00 2.10 2.20 0.0
4S 1 0 0.00 6.10 1.40 2.20 0.0
4P 2 1 0.00 0.00 2.20 2.30 0.0
4P 2 1 0.00 6.30 1.40 2.30 0.0
3D 3 2 10.00 0.00 1.50 1.80 0.0
3D 3 2 0.00 -0.60 1.50 1.80 0.0

```

config の修正
基本は config の修正だけでOK

zval があるときは hole
にしたぶん電子を抜く

擬ポテンシャル生成スイッチ

作成する擬ポテン
シャルの名前

★追加★

2) ホールを開ける (Zn_fch.in 作成)

configu 中に [Ar] とあるものは Ar コアなので省略形は止めてすべて書き下す

Ar = 1s2 2s2 2p6 3s2 3p6

config =
1s2 2s2 2p6 3s2 3p6 3d9.5 4s1.5 4p0.0



1sに core-hole

config =
1s1 2s2 2p6 3s2 3p6 3d9.5 4s1.5 4p0.0

Au 1s full core hole

config=
1s2 2s2 2p6 3s2 3p6
3d10. 4s2. 4p0.0



1sに core-hole

config=
1s1 2s2 2p6 3s2 3p6
3d10. 4s2. 4p0.0

Zn 1s full core hole

2) ホールを開ける (Zn_fch.in 作成)

3) ld1.x の実行

ld1.x < Zn_fch.in > out_ld1_Zn_fch.out

4) コアの wfc ファイルを作成

```
upf2plotcore.sh Zn.pbe-dn-kjpaw_psl.0.2.2_fch.UPF> Zn_fch.wfc
```

XANES の計算ステップ

- 1) 始状態コア波動関数**
- 2) 終状態(with core hole)のバンド計算**

コアホールに対応したSCF入力ファイルの用意

コアホールに対応した SCF 入力ファイルの作成

```
ntyp = 3,  
tot_charge = +1.0,  
ecutwfc = 25.0,  
occupations = tetrahedra,
```

擬ポテンシャルの種類を増やす
コアホールを開けた分だけ電子を増やす

```
/  
&electrons  
    mixing_beta = 0.7,  
    conv_thr = 0.01,  
/  
CELL_PARAMETERS {alat}  
    6.498000145000000  0.000000000000000000  0.000000000000000000  
   -3.249000072000000  5.6274331990000000  0.000000000000000000  
    0.00000000000000000  0.00000000000000000 10.413999557000000
```

ATOMIC_SPECIES

Zn 65.30000 Zn.pbe-dn-kjpaw_psl.0.2.2.UPF

Znc 65.30000 Zn.pbe-dn-kjpaw_psl.0.2.2_fch.UPF コアホール用の擬ポテ

O 15.90000 O.pbe-n-kjpaw_psl.0.1.UPF

ATOMIC_POSITIONS {crystal}

コアホール用の擬ポテを用いる吸収原子の指定

Znc 0.333333343000000 0.166666672000000 0.000000000000000000

Zn 0.333333343000000 0.166666672000000 0.500000000000000000

Zn 0.333333343000000 0.666666672000000 0.000000000000000000

xspectra.x

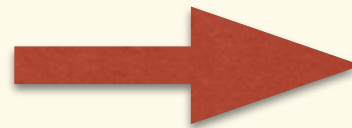
準備

fch 計算のフェルミレベルを入手

pw.x < *.in > **log**

```
13.5534 13.6576 13.8351 14.0306 14.2341 14.4544 14.5542 14.6956
14.9036 15.3621 15.4686 15.5358 15.7628 15.9921 16.0535 16.1957
16.2795 16.3338 16.4176 16.4894 16.5148
```

the Fermi energy is 7.9021 ev



この結果をRyに
直したものが必要

```
! total energy          = -4574.37657476 Ry
  Harris-Foulkes estimate = -4574.37807056 Ry
  estimated scf accuracy  < 0.00160992 Ry
```

```
total all-electron energy = -59166.639242 Ry
```

The total energy is the sum of the following terms:

計算ログ

PWgetfetmi.py

xanes_dipole or xanes_quadrupole

```
&input_xspectra
```

```
  calculation='xanes_dipole',
```

```
  prefix='pwscf',
```

```
  outdir='work',
```

```
  xonly_plot=.false.,
```

```
  xniter=1000,
```

```
  xcheck_conv=50,
```

```
  xerror=0.001,
```

Perform a full continuous
fraction calculation

control the Lanczos

Lanczos method の系数
を保存するファイル

```
  x_save_file='prefix.xspectra.sav',
```

```
  ef_r=0.574496,
```

★フェルミレベル(Ry単位)★

```
  xiabs=3,
```

```
  xepsilon(1)=1.0,
```

```
  xepsilon(2)=0.0,
```

```
  xepsilon(3)=0.0,
```

吸収原子の選択

★ATOMIC_SPECIES in pwscf input)★

ビームによる電場方向

```

/
&plot      ★(orentzian broadening, related to the core-hole lifetime )★
  xgamma=0.7, linewidth to be used in the spectrum [eV]

  xnepoint=1000, number of energy points in the plot of the XAS spectrum

  xemin=-10.0,
  xemax=30.0,

  terminator=.true., to use the terminator function for the continued fraction
  cut_occ_states=.true.,
/          the occupied states are smoothly cut from the plot

&pseudos
  filecore='Cu.wfc', ★吸収端★のcore状態(始状態) の波動関数
  ! r_paw(1)=3.2,
/

&cut_occ
  cut_desmooth=0.1,
  cut_step1=0.01,
/

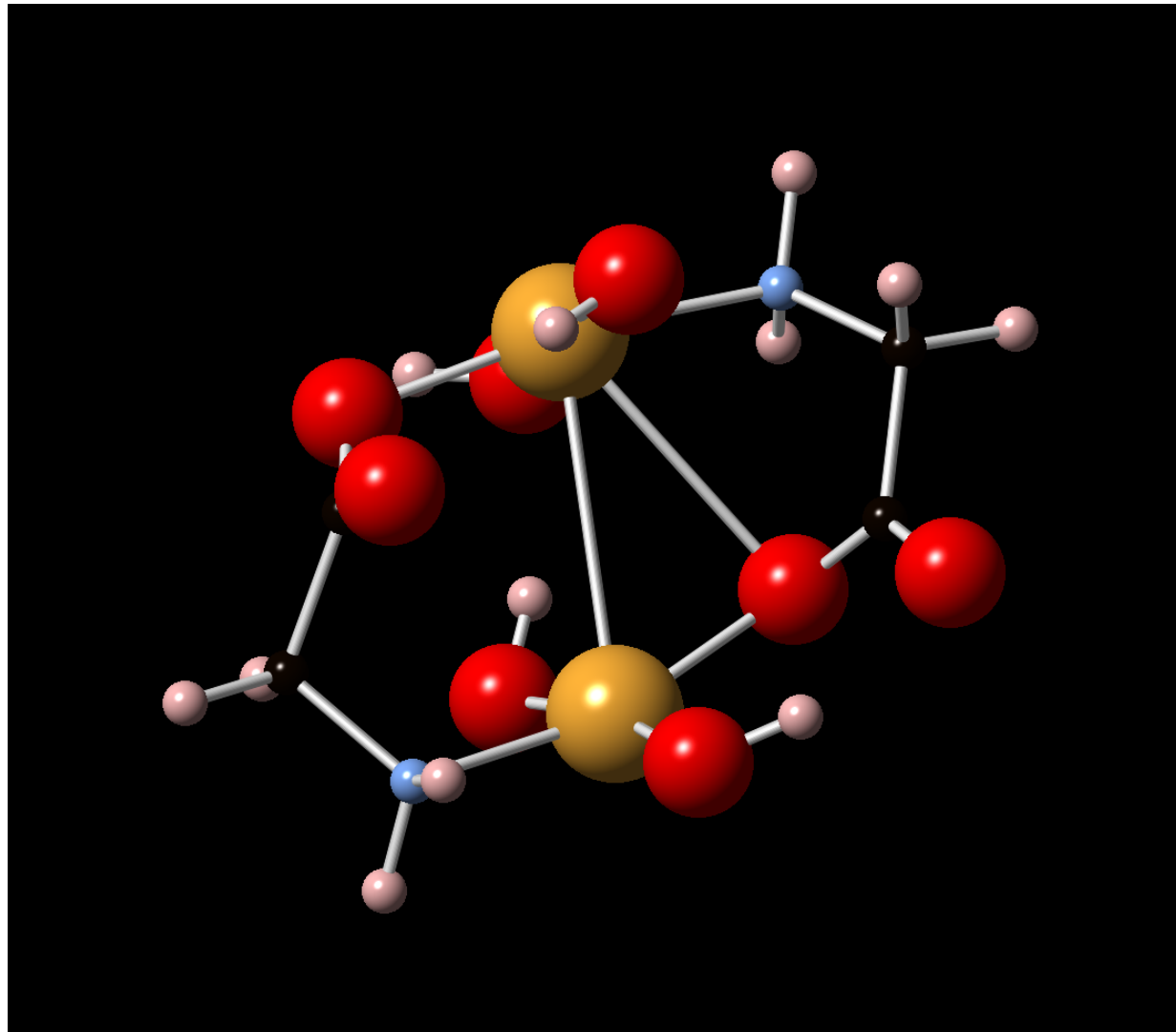
8 8 8 0 0 0      SCF計算で用いた K-point 情報

```

xspectra.x < xspectra.in

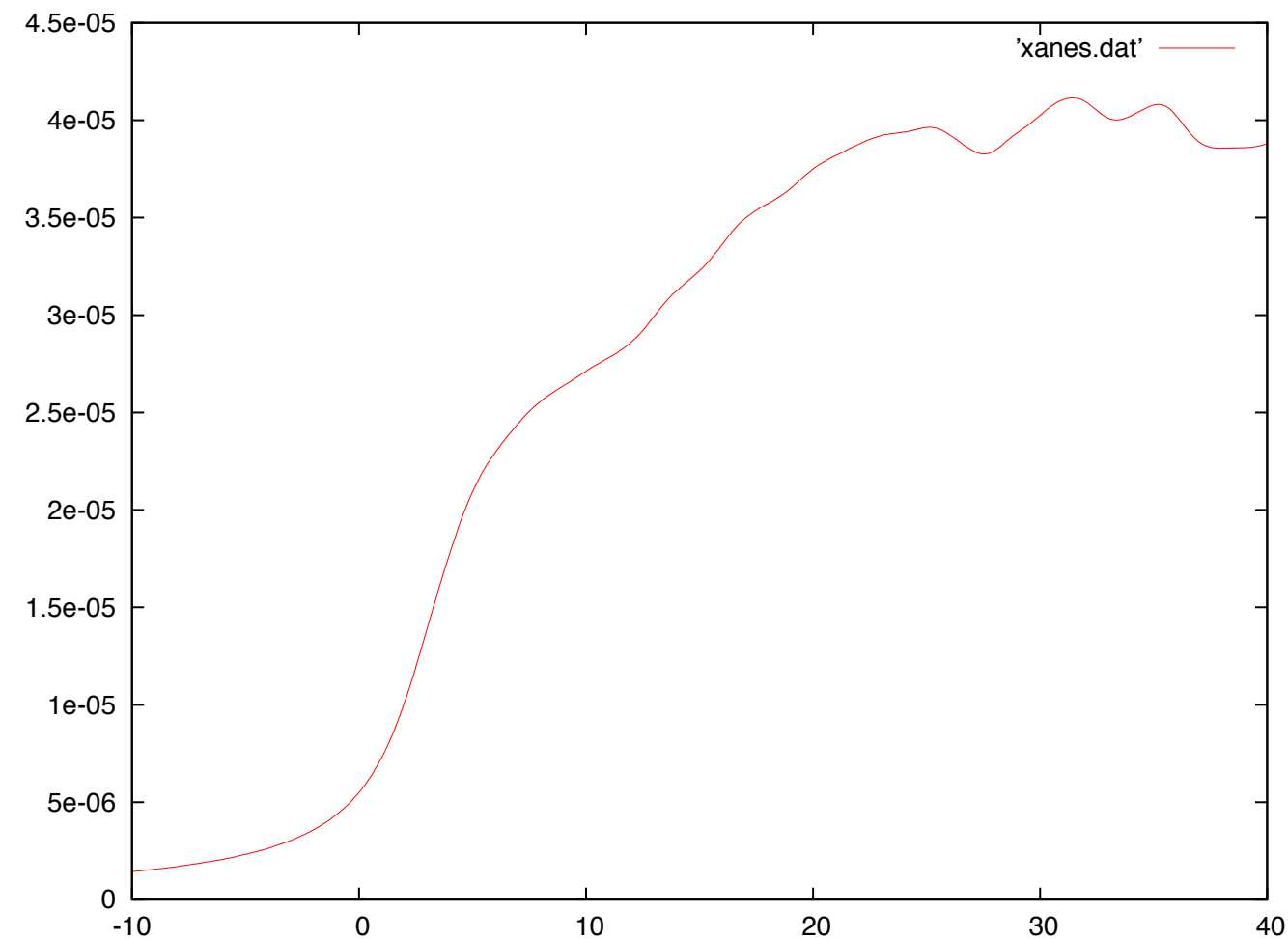
L1 端、計算テスト
(with core-hole)

Au2Glya2-4OH



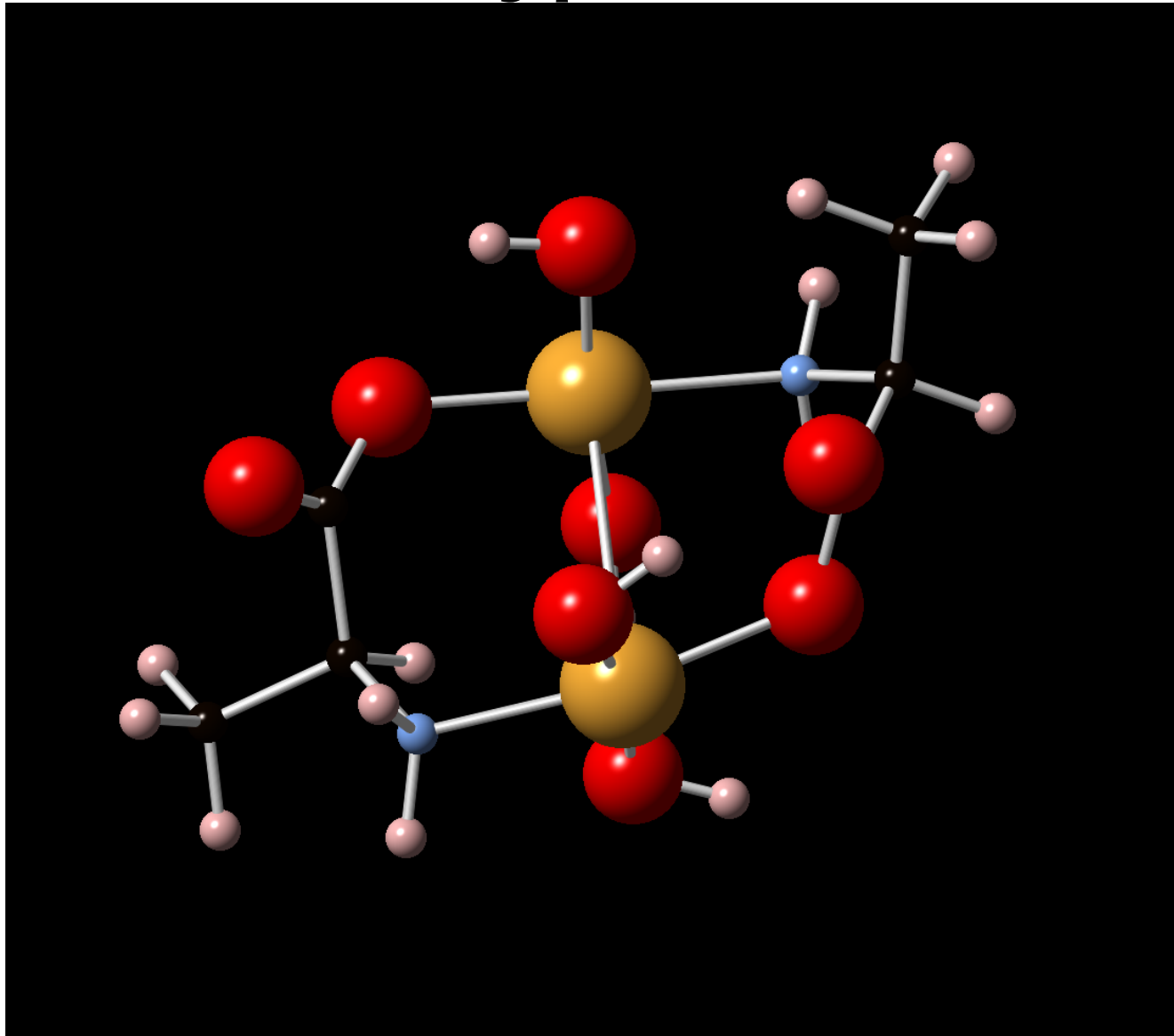
8Å

Au-L1 端 (FCH)



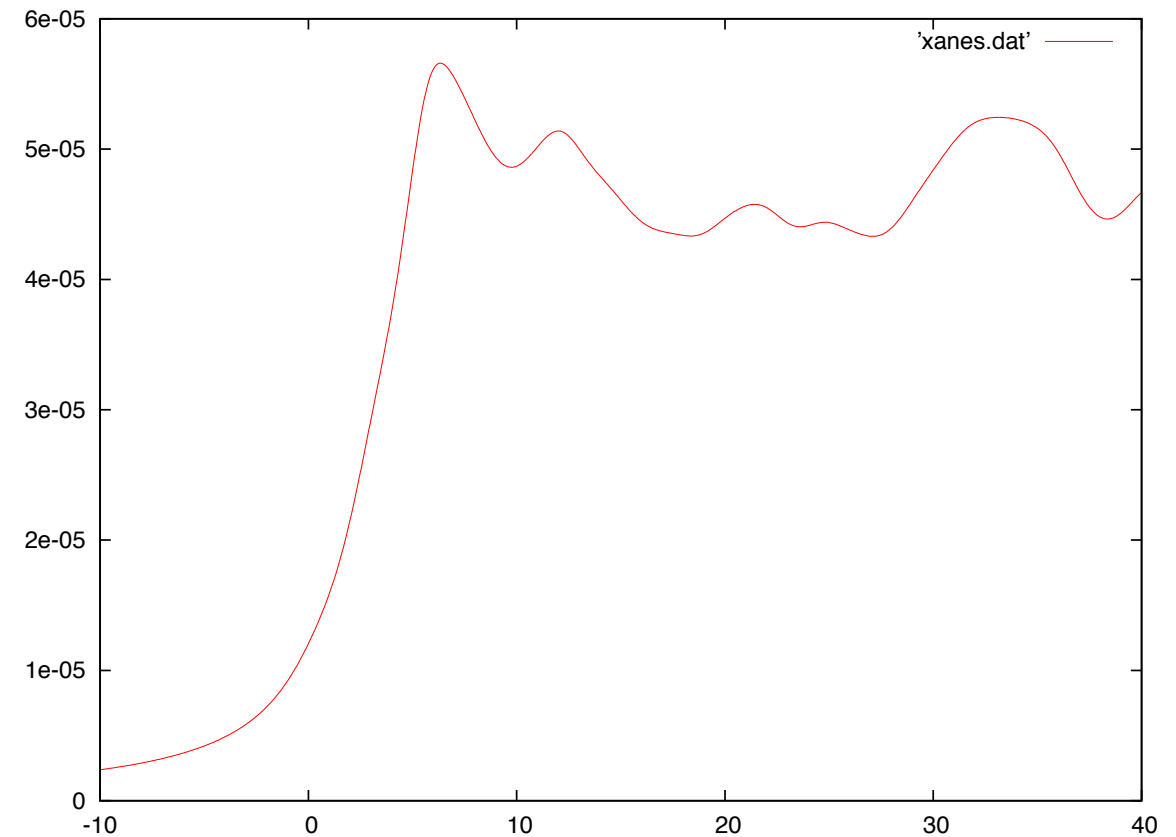
約18Åの箱を用意して計算

Au2a-ala-type2-4OH



11 Å

Au-L1 端 (FCH)

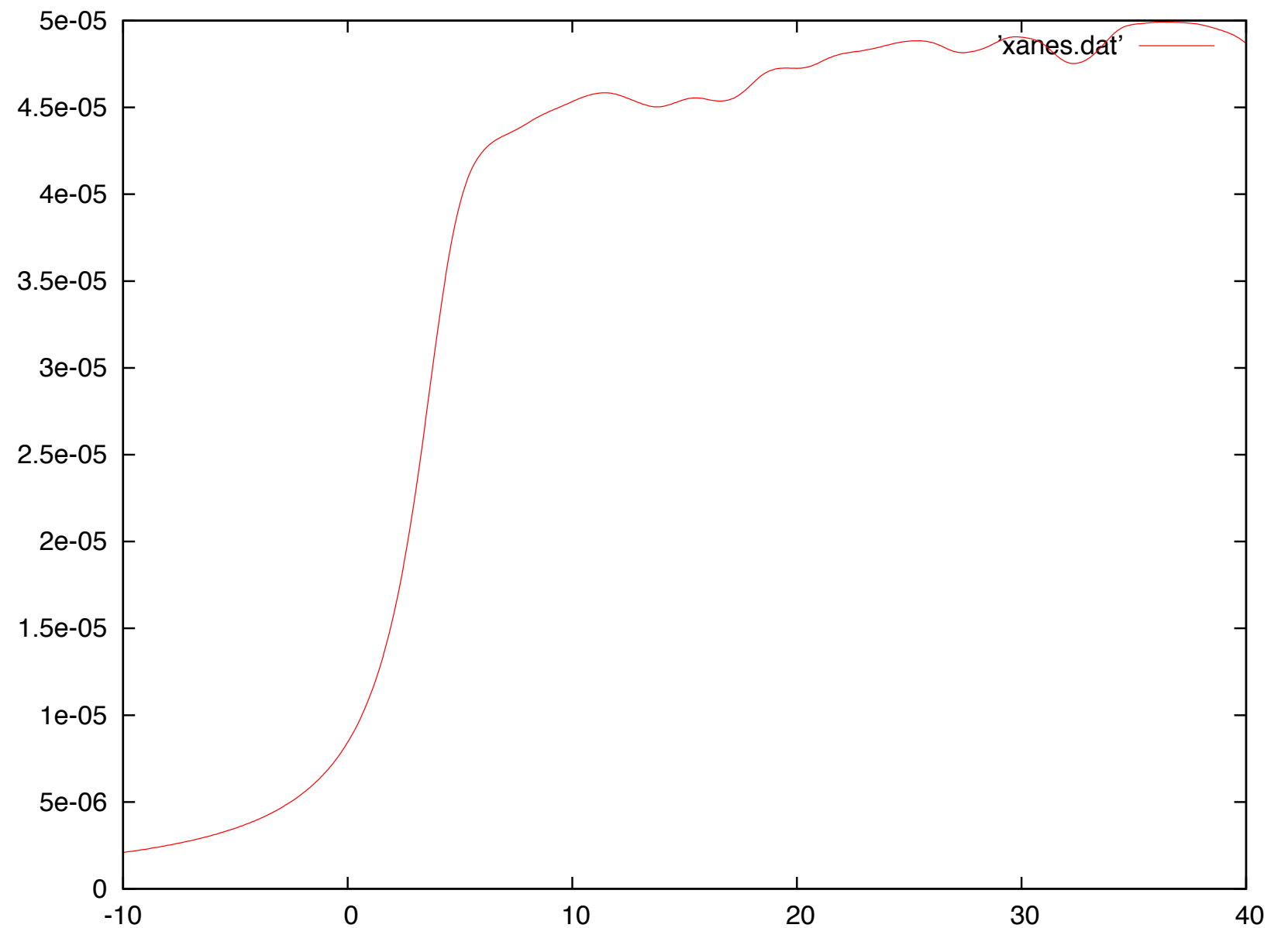


アホなミス

11 Åのまま計算してしまった
(分子をかなり稠密に配置)

約10Åの間隔を開けて、ほぼ孤立分子にした計算

Au-L1 端 (FCH)



解析用（LDOS/PDOS/XANES等の網羅的かつ詳細な描画）
のフレームワーク
とし各種クラス/関数・メソッドを作成

WIEN2k/VASP/OpenMX/CP2K/GPAW/HiLAPW/FHI-AIMS
で共通のフレームワーク

各種計算例

構造最適化

単位胞と内部座標の(同時)最適化

calculation = 'vc-relax'

&ions

/

&cell

/

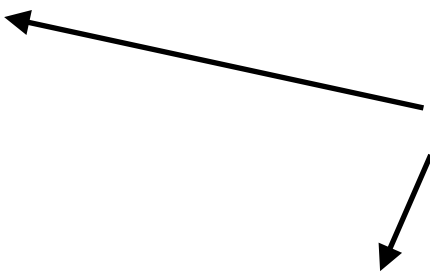
ion と cell に関するオプション指定枠を作成
(オプション指定をしなくても作成)

default → 単位胞の最適化 (格子定数の最適化)

内部座標の最適化

```
calculation = 'relax'  
etot_conv_thr = 1.d-5  
forc_conv_thr = 1.d-4
```

コントロールに入れる



default: relax-step は 50回まで (**nstep = 50**)

```
&ions  
/
```

ion に関するオプション指定枠を作成
(オプション指定をしなくても作成)

x,y,z を fixする (moveしない)

ATOMIC_POSITIONS {alat}

C 0.00 0.00 0.00 0 0 0

C 0.00 0.57735026918962576451 0.00 0 0 0

C 0.00 0.00 1.3

C 0.00 -0.57735026918962576451 1.3

```
Error in routine checkallsym (1):  
some of the original symmetry operations not  
satisfied
```

緩和計算で原子が動くことによって、**対称性が変わると**
計算が止まる！！

緩和計算で、緩和後（緩和過程）の対称性がわからない時は
`nosym = .true.`

構造最適化の初期値は VASP よりシビア

- 1) 原子が離れすぎていたら原子が動かないまま1回SCFで止まる
- 2) 計算途中で計算に用いる対称性を自動的に変更しないので、実質的に対称性なしで計算しなくてはいけないことが多い。

PWScf の結果を p4vasp 用の vasprun.xml へ変換

Au XAS

ld1.x

xspectra.x

PS Library

PWScf の PPライブラリ

USPP

(last updated Sept 10, 2014)

ANY FUNCTIONAL

ULTRASOFT

Apply Filter

ANY PP LIBRARY

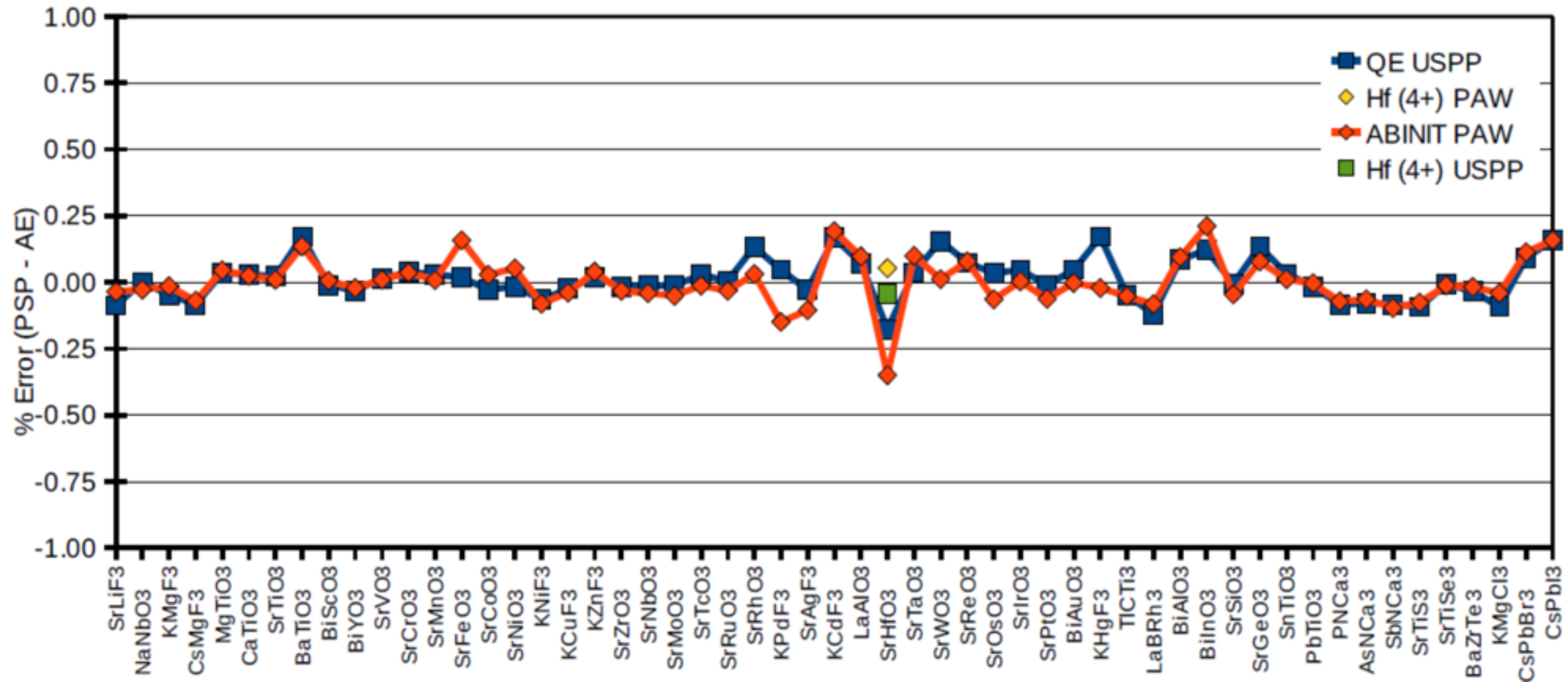
OTHER OPTIONS

1																	2	
H																	He	
3	4											5	6	7	8	9	10	
Li	Be											B	C	N	O	F	Ne	
11	12											13	14	15	16	17	18	
Na	Mg											Al	Si	P	S	Cl	Ar	
19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	
K	Ca	Sc	Ti	V	Cr	Mn	Fe	Co	Ni	Cu	Zn	Ga	Ge	As	Se	Br	Kr	
37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	
Rb	Sr	Y	Zr	Nb	Mo	Tc	Ru	Rh	Pd	Ag	Cd	In	Sn	Sb	Te	I	Xe	
55	56	57-70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86
Cs	Ba	*	Lu	Hf	Ta	W	Re	Os	Ir	Pt	Au	Hg	Tl	Pb	Bi	Po	At	Rn
87	88	89-102	103	104	105	106	107	108	109									
Fr	Ra	**	Lr	Rf	Db	Sg	Bh	Hs	Mt									

*	57 La	58 Ce	59 Pr	60 Nd	61 Pm	62 Sm	63 Eu	64 Gd	65 Tb	66 Dy	67 Ho	68 Er	69 Tm	70 Yb
**	89 Ac	90 Th	91 Pa	92 U	93 Np	94 Pu	95 Am	96 Cm	97 Bk	98 Cf	99 Es	100 Fm	101 Md	102 No

<http://www.quantum-espresso.org/pseudopotentials/>

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



PWScf の PPライブラリ

PAW

(last updated Sept 10, 2014)

ANY FUNCTIONAL ▾		PAW ▾		Apply Filter															
ANY PP LIBRARY ▾		OTHER OPTIONS ▾																	
1																		2	
H																		He	
3	4											5	6	7	8	9	10		
Li	Be											B	C	N	O	F	Ne		
11	12											13	14	15	16	17	18		
Na	Mg											Al	Si	P	S	Cl	Ar		
19	20		21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	
K	Ca		Sc	Ti	V	Cr	Mn	Fe	Co	Ni	Cu	Zn	Ga	Ge	As	Se	Br	Kr	
37	38		39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	
Rb	Sr		Y	Zr	Nb	Mo	Tc	Ru	Rh	Pd	Ag	Cd	In	Sn	Sb	Te	I	Xe	
55	56	57-70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	
Cs	Ba	*	Lu	Hf	Ta	W	Re	Os	Ir	Pt	Au	Hg	Tl	Pb	Bi	Po	At	Rn	
87	88	89-102	103	104	105	106	107	108	109										
Fr	Ra	**	Lr	Rf	Db	Sg	Bh	Hs	Mt										

*	57	58	59	60	61	62	63	64	65	66	67	68	69	70
Lanthanoids	La	Ce	Pr	Nd	Pm	Sm	Eu	Gd	Tb	Dy	Ho	Er	Tm	Yb
**	89	90	91	92	93	94	95	96	97	98	99	100	101	102
Actinoids	Ac	Th	Pa	U	Np	Pu	Am	Cm	Bk	Cf	Es	Fm	Md	No

<http://www.quantum-espresso.org/pseudopotentials/>

PSLibrary にある Quantum Espresso の PAW-PP

Pseudopotentials

Quantum Espresso Pseudopotentials - PSLibrary 0.3.1

Getting started with QE: Pseudopotentials

The following files hold the pseudopotentials in UPF format for over 60 elements in the periodic table, that are generated and generously shared by Prof. Dal Corso, SISSA, Italy.

These are the default pseudo potentials we use. Note they are PAW-PBE, so for solids, go with PBESOL package below

H																	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	La	Hf	Ta	W	Re	Os	Ir	Pt	Au	Hg	Tl	Pb	Bi	Po	At	Rn

PAW-PBE
結構計算できる

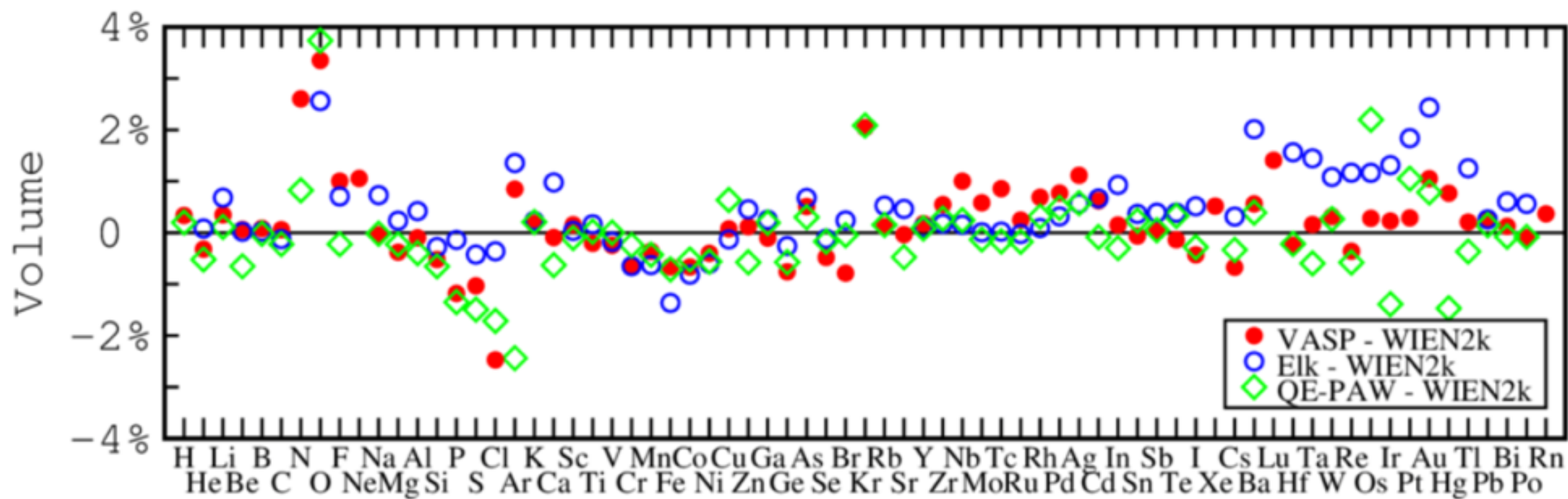
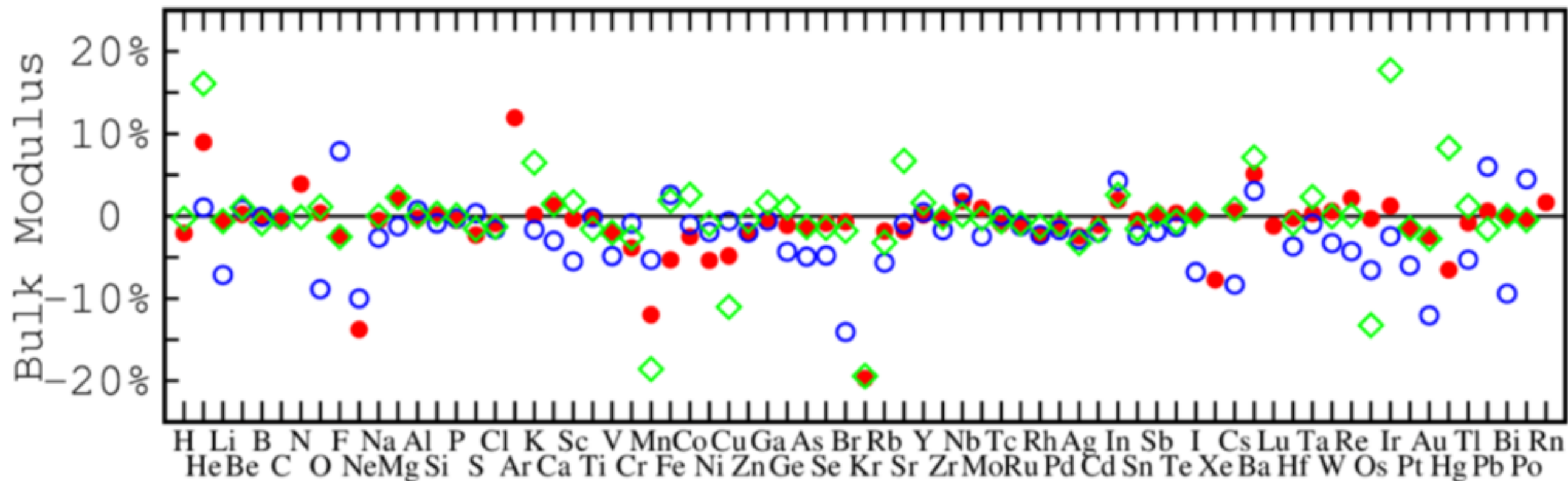
All pseudopotentials generated from PSLibrary 0.3.1

Download the compressed file for the exchange correlation type you wish to use. If you have no clue: PBE/PBESol are standard GGA choices for molecules and solids respectively; and if you need LDA instead go with "PZ".

(Beware, these are huge files. Each XC is approx 100MB, relativistic ones are approx. 200MB. Go to the Notes section below to learn more about the subfolders in each category.)

bp.0.3.1.tgz	rel-bp.0.3.1.tgz	pbe.0.3.1.tgz	rel-pbe.0.3.1.tgz
pbisol.0.3.1.tgz	rel-pbisol.0.3.1.tgz	pw91.0.3.1.tgz	rel-pw91.0.3.1.tgz
revpbe.0.3.1.tgz	rel-revpbe.0.3.1.tgz	pz.0.3.1.tgz	rel-pz.0.3.1.tgz
wc.0.3.1.tgz	rel-wc.0.3.1.tgz		

<http://theosrv1.epfl.ch/Main/Pseudopotentials>



Δ'_1 Factor between

QE-PAW 0.3.1 and WIEN2k

3.61 +/- 4.29 meV/atom

H																	He
1.2																	5.1
Li	Be											B	C	N	O	F	Ne
0.6	4.2											0.2	1.4	5.7	23.8	1.4	19.8
Na	Mg											Al	Si	P	S	Cl	Ar
0.1	1.5											2.5	4.3	9.2	10.1	14.3	12.1
K	Ca	Sc	Ti	V	Cr	Mn	Fe	Co	Ni	Cu	Zn	Ga	Ge	As	Se	Br	Kr
1.1	4.0	0.7	0.3	0.3	2.3	4.4	4.8	3.5	3.7	5.1	3.6	1.3	4.8	2.0	1.3	0.8	7.8
Rb	Sr	Y	Zr	Nb	Mo	Tc	Ru	Rh	Pd	Ag	Cd	In	Sn	Sb	Te	I	Xe
1.1	3.3	0.7	1.8	1.5	0.8	1.0	1.1	2.0	3.0	3.8	0.7	1.9	1.5	0.3	2.1	2.1	
Cs	Ba	Lu	Hf	Ta	W	Re	Os	Ir	Pt	Au	Hg	Tl	Pb	Bi	Po	At	Rn
2.4	2.9		1.5	4.0	1.7	3.8	0.6	4.3	7.0	5.2	9.4	2.2	0.7	0.5	0.6		

コアホールの計算が出来るポテンシャルが意外に少ない

ANY FUNCTIONAL ▾ ANY TYPE ▾ Apply Filter

ANY PP LIBRARY ▾ **HALF or ONE CORE-HOL ▾**

コアホールの計算に用いる

1 H																	2 He	
3 Li	4 Be											5 B	6 C	7 N	8 O	9 F	10 Ne	
11 Na	12 Mg											13 Al	14 Si	15 P	16 S	17 Cl	18 Ar	
19 K	20 Ca		21 Sc	22 Ti	23 V	24 Cr	25 Mn	26 Fe	27 Co	28 Ni	29 Cu	30 Zn	31 Ga	32 Ge	33 As	34 Se	35 Br	36 Kr
37 Rb	38 Sr		39 Y	40 Zr	41 Nb	42 Mo	43 Tc	44 Ru	45 Rh	46 Pd	47 Ag	48 Cd	49 In	50 Sn	51 Sb	52 Te	53 I	54 Xe
55 Cs	56 Ba	57-70 *	71 Lu	72 Hf	73 Ta	74 W	75 Re	76 Os	77 Ir	78 Pt	79 Au	80 Hg	81 Tl	82 Pb	83 Bi	84 Po	85 At	86 Rn
87 Fr	88 Ra	89-102 **	103 Lr	104 Rf	105 Db	106 Sg	107 Bh	108 Hs	109 Mt									